

Print modelling with generalised neighbourhoods

Stefaan Lippens, Wilfried Philips

Department of Telecommunications and Information Processing, Ghent University

Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium

tel: +32 9 264 34 12, fax: +32 9 264 42 95

Stefaan.Lippens@telin.ugent.be

Abstract—Printing is in its bare essence a binary process: putting ink on paper or not. To obtain the impression of gray values and colors one normally uses a pattern of small ink dots. The characteristics of these dots have a very important impact on the resulting print. It is well known that for today's print resolutions the actual density of a printed dot depends on neighbouring dots. The aim of print modelling is to quantify this relationship. Print models use the neighbourhood of a dot to map the pattern of neighbouring input pixels to an actual ink density. The neighbourhood size of traditional print models does not exceed 3×3 due to the exponential growth of the underlying mathematical problem [1, 8]. In the presented research we propose techniques to reduce the dimensionality in order to make it possible to tackle bigger neighbourhoods. For example, instead of using the whole pattern of the neighbourhood, the (binned) number of black dots in several dot groups are used as characteristic. The choice of the number and size of those dot groups is a degree of freedom that enables us to design bigger neighbourhoods while limiting the dimensionality of the mathematical problem.

We conducted some experiments based on printing a high number of periodic dot patterns with two laser printers. To model these printers, we tried more than 50 different neighbourhood group constellations, covering 1×1 up to 5×5 neighbouring dots and with their corresponding problem dimensionality ranging from 2 to over 800. Fitting the modelled dot reflectance to the measured reflectance of the printed patterns can be formulated mathematically as a constrained quadratic programming problem. We observed that low modelling errors can already be obtained with neighbourhoods of low dimensionality (order 20 to 30), which is even lower than the dimensionality of a traditional print model with 3×3 neighbourhood (with dimensionality 102).

Keywords—print modelling

I. INTRODUCTION

Traditional printing is in its bare essence a binary process: for every position on the paper one has two choices: putting ink or not putting ink. Despite this seemingly limitation, it is possible to obtain a broad pallet of gray scales and colors without the need for an individual ink for each

This research is funded by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

particular color. Based on the fact that the human visual system has low pass properties, the typical procedure is to distribute small spots of ink. When this pattern of spots is viewed from a sufficient distance, the spots will be unnoticeable and the human eye will only perceive a certain color. The precise properties of the perceived color will not only depend color of the spots, but also on their size and spacing. This makes it possible to obtain the impression of gray scale by only using black ink spots or produce color prints by only using a limited set of color inks (typically the four colors: cyan, magenta yellow and black).

Halftoning is the process that transforms an input image to a pattern of spots, in order to print the image. Several halftoning techniques exist, which differ in the properties of the spot pattern they generate: clustered versus dispersed spots, regular versus stochastic positioning, etc. In case of digital halftoning all these techniques construct the spot pattern out of so called “dots”, the smallest units of addressability of the printing device, conceptually equivalent to pixels. Current digital printing devices work on a scale of several hundred dots per inch. This means that the sizes of the used dots are several tens of micrometers. On this scale the dot is not what one expects from an ideal pixel: a perfect square or rectangle. As a first approximation, dots have a round form, for example caused by the spherical form of ink dots for ink based technologies or the circular laser beam profile for toner based technologies. Further deviation from the ideal square pixel model are problems like dot gain, caused by ink spread on the paper [1–3, 7–11], dot loss, indicating that some dots are bad or not accepted on the paper [3], interaction between dots [1], etc. It is clear that these mechanisms have an important impact on the final appearance of the printed image. Print modelling is therefore needed to compensate for these print distortions during the halftoning process.

Like already mentioned in the previous section, the most basic model of a dot is the circular dot model [3, 4, 6, 7, 9, 10], assuming that each printed dot is circular with a uniform distribution of ink. This geometric model does not account for interaction between neighbouring dots and therefore tends to fail for current and higher resolutions

[3]. At these resolutions the real density differs from what can be expected based on geometric grounds (overlap of circular areas).

Based on physical considerations of electrophotographic printing devices [5] proposes a dot model using probabilities of the toner particles being at a certain point on the photoconductive drum. Dot interaction is incorporated by an OR operation of the different probabilities of neighbouring dots. [3] identifies problems of this model related to undense toner packing seen at cluster centroids. He solves this problem by going one step deeper in the physics of the printing device. He uses the amount of light energy collected at the photoconductive drum and transforms this to probabilities of toner being at a position on the drum. With this model the probability of toner occurring for a particular dot is now directly affected by the concentration of dots in the surrounding area.

II. LOOK-UP BASED PRINT MODELS

The class of print models we will focus on in this paper make abstraction of the precise physics of the print process. These models try instead to model the ink or toner density with a look-up table in function of the neighbouring dot pattern at each dot. As a starting point we consider an input image¹ after halftoning: $I[x, y]$, which is a binary image defined in the discrete space corresponding with the printer addressability grid. After printing we get on paper an continuous-tone output *reflectance* image $\tilde{O}(\tilde{x}, \tilde{y})$ in continuous space (\tilde{x}, \tilde{y}) . As an important simplification we will use an equivalent discrete space output image $O[x, y]$ instead, which abstracts the continuous space effects in one reflectance per addressable cell of the printer. In short, the print model we want to deploy basically transforms the discrete binary image $I[x, y]$ to the discrete continuous tone image $O[x, y]$ without using continuous space concepts. For each pixel or dot² $[x, y]$ we will consider input pixels $I[x, y]$ and its neighbours $I[x + m, y + n]$ for $[m, n] \in W$ in order to determine the output $O[x, y]$. The set W defines the *neighbourhood* or window used for the print model. For example we can take a 3×3 neighbourhood ($W = [-1, 0, 1] \times [-1, 0, 1]$), in which $2^9 = 512$ different binary patterns P_n are possible. We define the print model by associating a reflectance p_n to each of these patterns. For clarity, we map the patterns to the reflectance of the *central* pixel $[x, y]$ of the neighbourhood, not to the mean reflectance of the whole pattern.

The reflectance p_n of the pattern P_n can be obtained

¹We will only consider gray scale input images here, the halftoned images are binary as consequence.

²In this discrete space discussion the concepts pixel and dot are equivalent.

with a direct and an indirect approach. The direct way, used in [1], involves printing the patterns P_n , scanning them with a high resolution scanner and calculating the reflectance of the central pixel. The main drawback of this approach is that precise registration is needed in order to catch the central pixel of the pattern. Furthermore, precise calibration of the scanning device is another difficult issue. The other approach, used in [8] and this paper, eliminates the difficulty of registration and calibration by only using macroscopic densitometric measurements. This technique will be explained further in section IV.

Given a certain neighbourhood W and a mapping of the associated patterns to reflectances, the transformation from input image $I[x, y]$ to output image $O[x, y]$ is simply done by considering the pattern around each pixel $[x, y]$ and looking up the associated reflectance.

The problem with this technique is the exponential grow of the lookup table in function of the size of neighbourhood W . If we expand the 3×3 neighbourhood of the example with one pixel in each direction, we have a 5×5 neighbourhood with more than 33 million possible patterns. On the other hand, some patterns can be considered the reflection or rotation of other patterns. If we assume that the reflection or rotation of a pattern makes no difference of the reflectance of the central pattern, it is possible to eliminate some entries in the lookup table. This technique would for example reduce the number of necessary 3×3 patterns to from 512 to 102. Regrettably, it would not make a 5×5 print model practically feasible to use.

III. GENERALISED NEIGHBOURHOODS

The research presented in this paper tries to eliminate more entries in the lookup table in order to make it feasible using neighbourhoods with a size exceeding 3×3 . The principles are based on a near field versus far field separation, like sometimes used in electromagnetism. Hereby we assume that individual pixel close to the central pixel of the pattern have a more important influence on the actual reflectance of the central pixel than individual pixels further away. On the other hand, the pixels further away are greater in number, which leads to a more important *combined* influence. Through sacrificing resolution on the pixel further away we want to limit the dimensionality of the underlying mathematical problem (see section IV). We use the following two techniques for this:

- *Grouping* pixels in a “neighbour group”. Up until now we used the whole binary pattern in the neighbourhood W to identify its state. Within a neighbour group we can limit the number of states by just looking at the number of black pixels in the group. For example, in a group of 5 pixels there are $2^5 = 32$ different binary patterns possible, but

TABLE I

SOME EXAMPLES OF GENERALISED NEIGHBOURHOODS AS A SET OF NEIGHBOUR GROUPS AND THEIR CORRESPONDING NUMBER OF PATTERNS AND SIGNATURES.

| | (A) | (B) | (C) | (D) | (E) |
|---|---|---|---|---|---|
| neighbour groups (with bin size between brackets if different from 1) | $\begin{bmatrix} 8 & 5 & 9 \\ 4 & 1 & 2 \\ 7 & 3 & 6 \end{bmatrix}$ | $\begin{bmatrix} 6 & 5 & 6 \\ 4 & 1 & 2 \\ 6 & 3 & 6 \end{bmatrix}$ | $\begin{bmatrix} 24 & 19 & 13 & 20 & 25 \\ 18 & 8 & 5 & 9 & 21 \\ 12 & 4 & 1 & 2 & 10 \\ 17 & 7 & 3 & 6 & 14 \\ 23 & 16 & 11 & 15 & 22 \end{bmatrix}$ | $\begin{bmatrix} & & 7 & & \\ & 6 & 5 & 6 & \\ 7 & 4 & 1 & 2 & 7 \\ & 6 & 3 & 6 & \\ & & 7 & & \end{bmatrix}$ | $\begin{bmatrix} & & 7 & & \\ & 6 & 5 & 6 & \\ 7 & 4 & 1 & 2 & 7 \\ & 6 & 3 & 6 & \\ & & 7 & & \end{bmatrix}$ <small>(3) (3) (3)</small> |
| the number of pixels in neighbourhood W | 9 | 9 | 25 | 25 | 25 |
| N , the number of possible patterns P_n | 512 | 512 | 33554432 | 33554432 | 33554432 |
| the number of signatures \mathbf{S}_n | 512 | 160 | 33554432 | 2720 | 960 |
| N_B , the number of basic signatures $\mathbf{S}_n^{(B)}$ | 102 | 60 | n/a | 1020 | 360 |

only 6 possibilities for the number of black pixels: 0, 1, 2, 3, 4 or 5.

- *Binning* the number of black pixels in a neighbour group. Using the previous example of a 5 pixel sized neighbour group will be the easiest to explain. If we join the possibilities for the number of black pixels per two, there are only 3 states (or bins) for the neighbour group left: “0 or 1 black pixel”, “2 or 3 black pixels” and “4 or 5 pixels”.

With these techniques we change the concept of a neighbourhood W from a set of pixels to a set of L neighbour groups w_l . Each neighbour group w_l covers one or more pixels and uses its own binning. If we now consider a binary pattern P_n , we count the number of black pixels in each neighbour group w_l to obtain the applicable bin $b_l(P_n)$. With those L values $b_l(P_n)$ we construct a vector \mathbf{S}_n , which we call the *signature* of the pattern P_n , because it gives a characterization of the pattern. Because of the grouping and the binning the number of possible signatures \mathbf{S}_n will be typically much smaller than the number of possible patterns P_n . Moreover, if we eliminate the doubles due to reflection and rotation like mentioned at the end of section II, we obtain an even more limited number of *basic signatures* $\mathbf{S}_n^{(B)}$, denoted by N_B .

Table I illustrates these presented concepts with some simple examples. Neighbourhoods (A) and (C) are the classic case of using the individual pixels as identification of the pattern (each neighbour group contains just one pixel) like discussed in section II and used in [8]. Neighbourhood (B) covers the same pixels as (A), but groups the pixels in the corners in one group, which reduces the number of signatures from $2^9 = 512$ to $2^5 \cdot 5 = 160$. The number of basic signatures is also reduced from 102 to 60. The reduction are even more striking with the 5×5 sized neighbourhood examples (C), (D) and (E). Neighbourhood (D) groups pixel 6 to 9 of (C) in one group, like in the 3×3 example, and also groups the pixels 10 to 25 of (C) in an

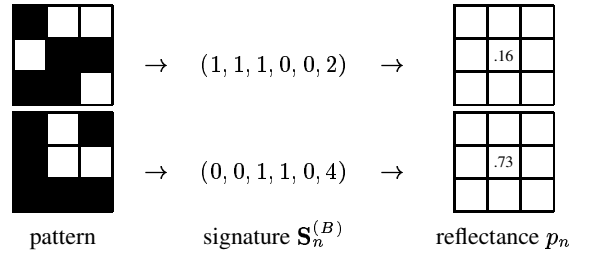


Fig. 1. Application of a printer model with signatures based on neighbourhood (B) from table I.

other group. This already reduces the number of signatures with more than 4 orders of magnitude. Neighbourhood (E) also uses binning on the surrounding neighbour group by grouping the possible number of black pixels per 3. This reduces the number of basic signatures to 360 which makes a 5×5 print model practically feasible.

Although the examples from table I span a square neighbourhood, this is not required. To make it possible to eliminate doubles due to reflection and rotation, only some symmetry around the central pixel is needed. Another variation is possible by introducing an offset of a halve pixel in both directions between the pixel grids of $I[x, y]$ and $O[x, y]$. This eliminates the intuitive mapping from pixels between $I[x, y]$ and $O[x, y]$, but makes it possible to use even sized neighbourhoods such as 2×2 and 4×4 .

IV. TRAINING/SOLVING THE MODEL

With the introduced concepts we can change, mutatis mutandis, the look-up based print model from section II. Instead of associating an effective reflectance p_n to the central pixel of a pattern P_n , we now associate a reflectance to the signature $\mathbf{S}_n^{(B)}$ of the patterns. The aim is here to obtain values for p_n in order to define our print model. In this section we will develop a mathematical and macroscopic based alternative to the microscopic reflectance measurements of p_n as used in [1].

The technique, also used in [8], only uses global densitometric measurements of a set of M printed test patches \hat{Q}_m . Each of these binary test patch is the periodic repetition of a Q_m , a small binary pattern of $w \times h$ pixels. With the densitometer we measure (without a need for registration) the effective reflectance \hat{q}_m of each printed test patch \hat{Q}_m . The size $m \times h$ should not be too big, in order to have enough repetition of the pattern in the measurement area of the densitometer. Moreover, $m \times h$ also should not be smaller than the coverage of the neighbourhood. In this case we can assume that the measured global reflectance of the printed test patch \hat{Q}_m is the same of the mean reflectance of the printed pattern Q_m . This is important because, with the help of the print model, we can express this mean reflectance as a function of the (yet unknown) p_n . For each pixel $[x, y]$ of binary pattern Q_m we explore its neighbourhood (at borders Q_m should be extended periodically) to obtain the basic signature $\mathbf{S}_{n(x,y;m)}^{(B)}$ and the associated reflectance $p_{n(x,y;m)}$ of that pixel. The modelled mean reflectance of pattern Q_m is just the mean of these values $p_{n(x,y;m)}$, so we get:

$$\frac{1}{wh} \sum_{x=1}^w \sum_{y=1}^h p_{n(x,y;m)} = \hat{q}_m \quad \text{for } m = 1, 2, \dots, M$$

This set of equations can be simplified by counting the number of times each signature $\mathbf{S}_n^{(B)}$ (and its associated reflectance p_n) occurs in each pattern Q_m . If we call this number of times $a_{m,n}$, we get:

$$\frac{1}{wh} \sum_{n=1}^{N_B} a_{m,n} p_n = \hat{q}_m \quad \text{for } m = 1, 2, \dots, M$$

If we define a $M \times N_B$ matrix C with $C_{m,n} = \frac{a_{m,n}}{wh}$, this expression can be written as a matrix equation

$$C\mathbf{p} = \hat{\mathbf{q}} \quad (1)$$

Two problems arise when one wants to solve this equation directly, for example by using N_B test patches (making C a square matrix) and try to invert it. First of all, there is a certain noise on the measurements $\hat{\mathbf{q}}$, due to variations of the printing process itself and measuring noise. This suggests a more statistical approach, in which M , the number of printed and measured patches, should be bigger than N_B , the number of parameters in the print model. A second more important and radical problem is the occurrence of linear dependencies between the rows of C . For all (except some trivial small) neighbourhoods used in the experiments we observed relationships between the occurrence of signatures and patterns, independent of the

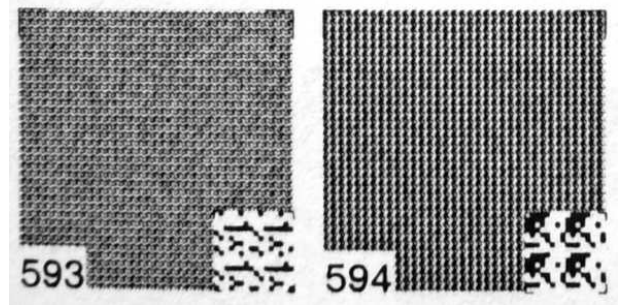


Fig. 3. Close-up of two printed patches. In the lower right corner of each patch an enlarged version is also printed depicting Q_m four times. The actual size of the patches is 1.5 cm \times 1.5 cm

used pattern Q_m . See Fig. 2 for a simple example. The mathematical consequence of this is that the matrix C is not invertible.

As done in [8], we solve equation 1 by looking for a \mathbf{p} that minimizes the quadratic error between $C\mathbf{p}$ and $\hat{\mathbf{q}}$. The linear dependencies between row of C suggest that extra constraints are needed, such as the reflectance being between r_{min} , the minimum reflectance possible (the reflectance of a totally black patch) and r_{max} , the reflectance of unprinted paper. Doing so, we end up with a quadratic program:

$$\begin{aligned} &\text{minimize } (C\mathbf{p} - \hat{\mathbf{q}})^T (C\mathbf{p} - \hat{\mathbf{q}}) & (2) \\ &\text{with } r_{min} \leq p_n \leq r_{max} \quad \text{for } n = 1, 2, \dots, N_B & (3) \end{aligned}$$

In short, given a set of M binary patterns Q_m , we can count the occurrences of each signature $\mathbf{S}_n^{(B)}$ and construct matrix C . We also print patches build of the patterns Q_m and measure their reflectance $\hat{\mathbf{q}}$. The solution of the quadratic program (2,3) is the N_B -dimensional vector \mathbf{p}^* and defines the print model. In the general case, the residual error $\mathbf{e} = C\mathbf{p}^* - \hat{\mathbf{q}}$ will not be zero. For being able to compare different neighbourhoods we will use the root mean squared error $\text{RMSE} = \sqrt{\frac{\mathbf{e}^T \mathbf{e}}{M}}$. This error has three contributors: model incompleteness, variability of the printing process and measurement noise. The first two are the interesting ones, whereas the measurement noise should be minimal.

V. EXPERIMENTAL RESULTS

We used the described method to model two printers at our lab, called “verdana” and “zapf”. Both are 600 dpi laser printers, verdana is a HP LaserJet 4300 and zapf is a Nashuatec DSc224 copier/printer. We printed two different patch sets \hat{Q}_m and measured the density with a X-Rite spectrodensitometer to obtain the global reflectance



| | $\# \begin{array}{ c c } \hline \blacksquare & \blacksquare \\ \hline \end{array} + \# \begin{array}{ c c } \hline \blacksquare & \square \\ \hline \end{array} + \# \begin{array}{ c c } \hline \square & \blacksquare \\ \hline \end{array} = \# \begin{array}{ c c } \hline \blacksquare & \square \\ \hline \end{array} + \# \begin{array}{ c c } \hline \square & \square \\ \hline \end{array} + \# \begin{array}{ c c } \hline \square & \blacksquare \\ \hline \end{array}$ | $\# \begin{array}{ c c } \hline \blacksquare & \blacksquare \\ \hline \end{array} + \# \begin{array}{ c c } \hline \square & \square \\ \hline \end{array} = \# \begin{array}{ c c } \hline \square & \blacksquare \\ \hline \end{array} + \# \begin{array}{ c c } \hline \square & \square \\ \hline \end{array}$ |
|---|---|--|
| Q_1 :  | $2 + 2 + 0 = 2 + 1 + 1$ | $1 + 1 = 0 + 2$ |
| Q_2 :  | $1 + 2 + 2 = 1 + 0 + 4$ | $1 + 0 = 0 + 1$ |

Fig. 2. Simple example of the relationships between occurrences of signatures (for simplicity the signatures are just patterns without elimination due to reflection and rotation).

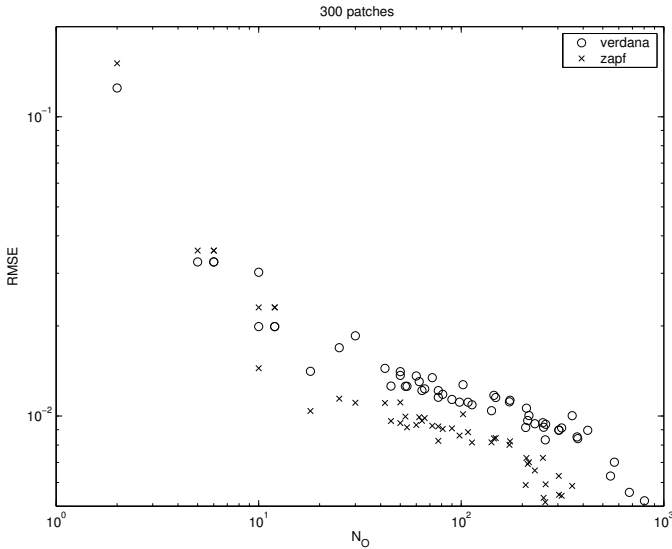


Fig. 4. RMSE of print model in function of N_O for verdana and zapf using the patch set with 300 patches.

of each patch. The first set consists of 300 patches \hat{Q}_m , each build by periodically repeating a 7×7 random pattern Q_m and was measured for both verdana and zapf. The second set consists of 1000 patches \hat{Q}_m , this time build with 8×8 random patterns, and was only measured for verdana. Fig. 3 shows two patches of this second patch set. The random patterns patterns Q_m were generated by thresholding uniform distributed white noise. The threshold was chosen different for each pattern to try avoiding that certain patterns P_n or basic signatures $\mathbf{S}_n^{(B)}$ would not occur in a whole set of test patterns. Unfortunately, this goal was not always achieved since for neighbourhoods with a rather high N_B some basic signatures did not occur in the patch sets after all. In the experiments we conducted however it was possible to ignore the signatures that did not occur. In the following discussion N_O will denote the number of occurred basic signatures as N_O . We used more than 50 neighbourhood configurations. The coverage of these neighbourhoods ranges from 1×1 to 5×5 pixels and their N_O ranges from 2 to 803.

Fig. 4 shows the RMSE of the printmodel for the different neighbourhood configurations in function of their

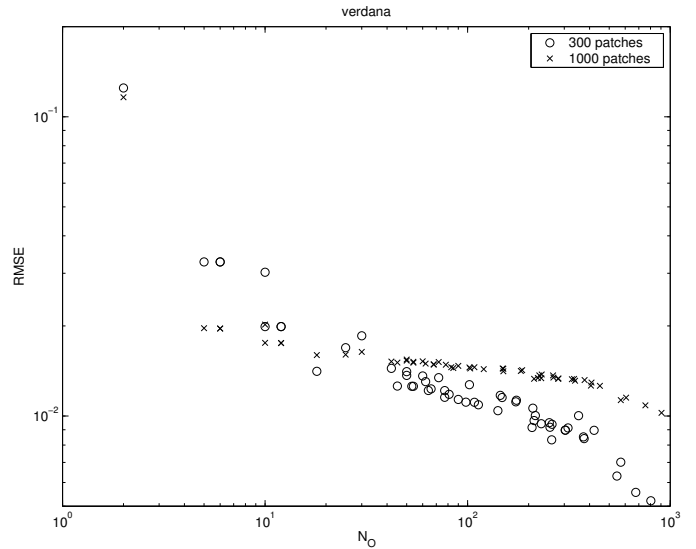


Fig. 5. RMSE of print model in function of N_O for verdana using both patch sets (300 and 1000 patches).

number of basic signatures, in case the 300 patch set and for both printers. It can be clearly seen that the RMSE for zapf is systematically lower than the RMSE of verdana. This assures that the error components related to the model incompleteness and the variability of the printing process are bigger than the error component related to measurement noise. We can conclude this because the measurement noise is the same for both printers. Moreover we have no reason to believe that zapf, has no print variability or could be perfectly modelled. Furthermore, as expected, the RMSE shrinks when N_O increases, but the rate is rather small. At a N_O of around 30 the RMSE is already around 10^{-2} , meaning that the print model is on average 1 percent wrong about the reflectance at that point, and decreases not very much for higher N_O . This is an indication that for the used resolution (600 dpi) there is not much to gain by using neighbourhood constellations of higher dimensionality. Another interpretation is that the results obtained with near field/far field separation are comparable with the results of the full descriptive model of [8].

In Fig. 5 we only consider verdana, for both patch sets (300 and 1000 patches). The behaviour of RMSE for the

TABLE II

SOME EXAMPLES OF NEIGHBOURHOOD CONSTELLATIONS WITH THEIR N_B , N_O AND RMSE OF THE CORRESPONDING PRINT MODEL FOR VERDANA, TRAINED WITH THE 1000 PATCHES. THE NUMBER IN THE PIXELS DENOTE THE NEIGHBOUR GROUP THEY ARE IN. IF THE BIN SIZE DIFFERS FROM ONE, IT IS DISPLAYED BETWEEN PARANTHESES.

| name | MEZEL | YOCEM | FIFUX | XEKUK | SEWON | ZETEQ | KABAH | YUHUL |
|-------|--------|-------|-------|-------|-------|-------|-------|-------|
| | | | | | | | | |
| N_B | 2 | 6 | 54 | 78 | 12 | 102 | 50 | 90 |
| N_O | 2 | 6 | 54 | 78 | 12 | 102 | 50 | 90 |
| RMSE | 11.61% | 1.96% | 1.51% | 1.48% | 1.75% | 1.45% | 1.53% | 1.47% |

| name | GUGEC | HUWOY | JETIL | CIDOS | GUXID | VOFOZ |
|-------|-------|-------|-------|-------|-------|-------|
| | | | | | | |
| N_B | 18 | 306 | 108 | 300 | 72 | 200 |
| N_O | 18 | 280 | 103 | 281 | 68 | 186 |
| RMSE | 1.60% | 1.33% | 1.44% | 1.34% | 1.49% | 1.42% |

case of 1000 patches is even more flat in function of N_O and worse than the case of 300 patches. We assume that this is caused by the augmented influence of print variability, due to the higher number of patches. We also observe that the points in Fig. 5 of the 1000 patches experiment are less scattered than in the 300 patches case. Because the 1000 patches experiment is statistically more important, we can conclude from this that the type of dimensionality reduction (grouping or binning) has very little influence on the print model accuracy.

Table II gives some examples of neighbourhood constellations³ with their N_B , N_O and the RMSE of the experiment with verdana and the 1000 patches. Like in Fig. 5, it illustrates that bigger N_O values correspond with lower RMSE. Consider for example the set neighbourhoods MEZEL, YOCEM, SEWON, and ZETEQ, all traditional neighbourhoods, in the sense that they do not use neighbour grouping or binning. The first neighbourhood, MEZEL, is a simple 1×1 printer model with 2 states and results in a RMS error of 11.6%. YOCEM, an 2×2 neighbourhood with only 6 states, offers already an error of only 1.96%. Such a huge difference is not repeated. Compare for example YOCEM and SEWON, the doubling in dimensionality only reduces the RMSE with 0.21%. The same order of RMSE-reduction can be seen

³The seemingly silly names of the neighbourhoods are in fact compact hash representations we used to make it easier to handle these different neighbourhoods.

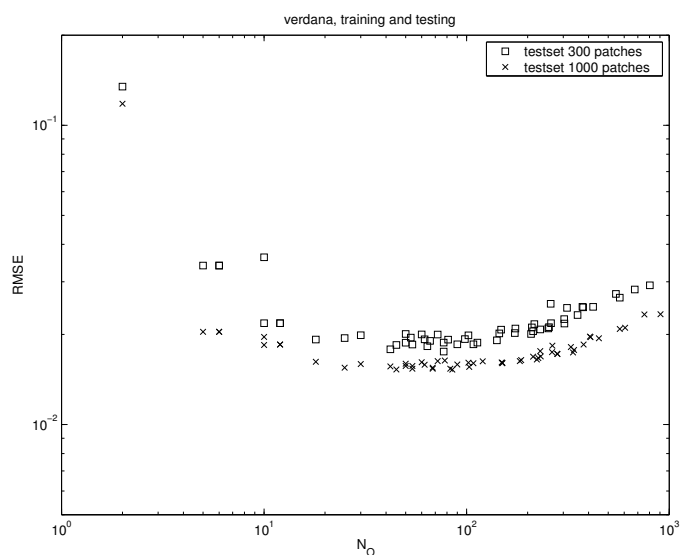


Fig. 6. RMSE of the test set in function of N_O for verdana using both patch sets (300 and 1000 patches), each time split in a training and test set.

from SEWON to ZETEQ, but it involves a more than four times bigger dimensionality. The usage and benefits of grouping can be seen by inspecting each of the neighbourhood sets {YOCEM, FIFUX, XEKUK}, {ZETEQ, KABAH, YUHUL} or {ZETEQ, GUGEC, HUWOY}. Binning is illustrated in neighbourhoods sets {GUGEC, JETIL, GUXID} and {KABAH, CIDOS, VOFOZ}.

In a last experiment we split the patch set in two equal sized subsets. With one set, the training set, we train the print model and with the other set, the test set, we evaluate the trained model. Fig. 6 shows the RMSE of the test set for both the 300 patches case and the 1000 patch case. Now the RMSE of the 300 patches case is worse than the RMSE for the 1000 patches case. This can be attributed to the fact that the print models generalise better for 1000 patches than for 300 patches. Also, from a certain N_O the RMSE increases again for both cases. This to can be explained by worse generalisation when the number of parameters to train (N_O) becomes to big compared to the number of measurements (M , 150 and 500 respectively).

VI. CONCLUSIONS

We presented a generalised neighbourhood framework to characterise the binary neighbourhood of a pixel with reduced dimensionality. This made it possible to explore look-up based print models spanning a neighbourhood that exceeds 3×3 . The entries in the print model's look-up table were obtained indirectly based on global macroscopic measurements and solving a quadratic program. We used this technique to try different neighbourhood constellations with different pixel coverages, group sizes and bin sizes. We used the residual error as a metric to compare different neighbourhood constellations. For the used print technology (laser printing at 600 dpi) we observed that descent print modelling can be obtained with low dimensional neighbourhood constellations, comparable to the results of the traditional 3×3 neighbourhood. We also observed a clear trade-off between print model accuracy and print model dimensionality, in which the type of dimensionality reduction (grouping or binning) has little importance.

REFERENCES

- [1] Farhan A. Baqai and Jan P. Allebach. Halftoning via direct binary search using analytical and stochastic printer models. *IEEE Transactions on Image Processing*, 12(1):1–15, 2003.
- [2] P. Kristiansson, C. M. Nilsson, H. Busk, L. Malmqvist, M. Elfman, K. G. Malmqvist, J. Pallon, K.A. Sjöland, R.J. Utui, and C. Yang. Optical dot gain on newsprint determined with the lund nuclear microprobe. *Nuclear Instruments and Methods in Physics Research*, 130(1-4):303–307, July 1997.
- [3] Daniel L. Lau and Gonzalo R. Arce. *Modern Digital Halftoning*. Marcel Dekker, 2001. ISBN 0-8247-0456-8.
- [4] Daniel L. Lau, Arif M. Khan, and Gonzalo R. Arce. Minimizing stochastic moire in frequency-modulated halftones by means of green-noise masks. *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, 19(11):2203–2217, 2002.
- [5] Q. Lin and J. Wiseman. Impact of electrophotographic printer dot model on halftone image quality. In *SID93 Conference Digest*, pages 147–150, Seattle, Washington, May 1993.
- [6] Thrasyvoulos N. Pappas. Model-based halftoning of color im-

- ages. *IEEE Transactions on Image Processing*, 6(7):1014–1024, July 1997.
- [7] Thrasyvoulos N. Pappas, Jan P. Allebach, and David L. Neuhoff. Model-based digital halftoning. *IEEE Signal Processing Magazine*, 20(4):14–27, July 2003.
- [8] Thrasyvoulos N. Pappas, Chen-Koung Dong, and David Neuhoff. Measurement of printer parameters for model-based halftoning. *Journal of Electronic Imaging*, 1993.
- [9] Thrasyvoulos N. Pappas and David L. Neuhoff. Printer models and error diffusion. *IEEE Transactions on Image processing*, 4(1):66–80, January 1995.
- [10] Thrasyvoulos N. Pappas and David L. Neuhoff. Least-squares model-based halftoning. *IEEE Transactions on Image Processing*, 8(8):1102–1116, Augustus 1999.
- [11] Li Yang, Sasan Gooran, and Björn Kruse. Simulation of optical dot gain in multichromatic tone production. *Journal of Imaging Science and Technology*, (45):198–204, 2001.