Coding efficiency of multi-ring and single-ring differential chain coding for telewriting application

J.Andrieux and G.Seni

Abstract: The recent availability of small personal digital assistants (PDAs) with a touchscreen and communication capabilities has been an influential factor in the renewed interest in telewriting, a technique for the exchange of handwritten information through telecommunications means. In this context, differential chain coding algorithms for compression of the handwritten ink are revisited. In particular, it is shown that the coding efficiency of multi-ring differential chain coding (MRDCC) is not always better when compared to single ring differential chain coding (DCC), as previously suggested. These algorithms were tested on over 300 handwritten messages using a relative compactness criterion and a per-length distortion measure. The probabilities of relative vectors in MRDCC and DCC are related, an expression for relative compactness in the MRDCC case is introduced, and the application of Freeman's criteria for the selection of the appropriate code for a family of curves is illustrated.

1 Introduction

The recent availability of small keyboardless personal digital assistants (PDAs) and Internet appliances (IAs) with a touchscreen and communication capabilities has been an influential factor in the renewed interest in telewriting [1], a method originally proposed for real-time encoding, transmission, and reproduction of handwritten data generated by dynamically tracing the movements of an electronic pen. The pen trajectory, or electronic 'ink', is usually given as a time-ordered sequence of x, y coordinates. Two-way transmission of electronic ink, possibly wireless, offers PDA/IA users a compelling new way to communicate. Users can draw or write with a stylus on the device's screen to compose a note in their own handwriting. Such an ink note can then be addressed and delivered to other PDA/IA users, Internet email users, or fax machines. The recipient views the message as the sender composed it, including text in any mix of languages and drawings. With this application in mind, we revisit differential chain coding algorithms for compression of the electronic ink.

Chain coding is a well known quantisation technique for line drawings, where the curve trajectory is represented by recording only transitions between successive points in a regular lattice, not absolute coordinates [2]. The quantised ink is thus represented by a sequence $\{d_i\}$ of vector displacements. Differential chain coding (DCC) [3], or chain difference coding [4], is an encoding technique based on Huffman encoding of the sequence of differences $\{\delta_i = d_i - d_{i-1}\}$. That is, DCC exploits the fact that in a smooth curve, $\{d_i\}$ does not change a great deal from one sample to the next. This means that the dynamic range and the variance of the sequence of differences $\{\delta_i\}$ are significantly smaller than those of the $\{d_i\}$ sequence.

It is also well known that with standard chain coding, the shape and the dimension of the quantisation lattice, or quantisation ring, define to a large extent the accuracy and the efficiency of the representation [5]. In handwritten text and sketches it is common to find regions of largely varying speed of writing and scale, and thus differences in radius of curvature. Therefore, it is expected that the efficiency of the representation can be increased by adapting the dimension of the quantisation ring to the pen-speed and/or radius of curvature. This idea led to the development of the generalised chain codes (GCC), where a set of concentric quantisation rings is used. The minimum ring size can be selected to conform to the minimum expected radius of curvature to be preserved by the representation, and the selection of the other rings depends on the distribution of the radii of curvature [6].

GCC quantisation can also be combined with any differential encoding technique to achieve high compression efficiency. One such combination is termed multi-ring differential chain coding (MRDCC) [7]. Under MRDCC, the quantised ink is represented with a sequence of 'relative' and 'absolute' vectors. A relative vector corresponds to a two-bit encoding of a small difference δ_i, e.g. $|\delta_i| \leq 1$. An absolute vector is a fixed-word encoding of a vector displacement d_i in all other cases. MRDCC thus avoids variable-wordlength encoding, the motivation being algorithmic simplicity and resilience to transmission errors. These design characteristics make MRDCC attractive for our intended application.

Johannessen *et al.* [8] analysed the performance of MRDCC using a theoretical model, which allowed them to conclude that the use of multiple rings always resulted in the highest coding efficiency. We evaluated this result on a database of over 300 ink images corresponding to different message scenarios. We found that a single ring code can sometimes outperform a multiple ring code. We identified

IEE Proceedings online no. 20010443

DOI: 10.1049/ip-vis:20010443

Paper first received 29th August 2000 and in revised form 14th March 2001

J. Andrieux is with Eurecom Institute, 2229 Route des Crêtes, 06904 Sophia-Antipolis, France

G. Seni is with Motorola Human Interface Labs, 3145 Porter Drive, Palo Alto, CA 94304, USA

an invalid assumption in their model about the relationship between relative vector probabilities in MRDCC and DCC.

Other extensions of DCC are reported in the literature. Two such extensions are termed enhanced differential chain coding (EDCC) [9], and conditional differential chain coding (CDCC) [10]. EDCC suggests increasing the accuracy of a DCC code, at the expense of some efficiency, by increasing the number of discretisation points in the ring. CDCC suggests increasing the efficiency of a DCC code, at the expense of added complexity, by modelling second-order directional changes.

2 Overview of GCC quantisation

The generalised chain code, introduced by Freeman and Saghri [4, 6], is a multi-ring quantisation procedure intended to reproduce curving lines with higher smoothness and precision than with single ring codes, at the expense of some additional complexity.

A particular GCC quantiser is defined by specifying the number of rings, the nodes along each ring, and the link gate sets (LGS). More precisely:

Rings. Given positive integers m, N_1, $N_2, \ldots, N_m$, with $N_1 < N_2 < \cdots < N_m$, the $(N_1, \ldots, N_m)$-chain code uses m concentric square rings. N_i is the 'order' of the ith ring.

Nodes. In a standard ring of order N, there are $8N$ nodes uniformly placed around the boundary of a square lattice with sides of length $8N\tau$, where τ is the grid size (in pixels). In a non-standard configuration a subset of these nodes can be non-uniformly arranged.

LGS. A vector directed from the centre of a ring to to any of its nodes is called a 'link'. A link gate set is a decision region which provides means for selecting the links that best approximate the curve. Two different kinds of decision region have been proposed: triangular [6], and parallel [8, 11]. For a given node $Q_{(m)}$ on the N_m-ring, let $Q_{l,(m)}$ and $Q_{r,(m)}$ be its left and right neighbour points along the ring at distance $\tau/2$. Then:

- triangular tolerance band: lines are drawn from the centre of the rings to the midpoints $Q_{l,(m)}$ and $Q_{r,(m)}$. These lines form a 'cone' that contains the link ending at $Q_{(m)}$. The intersection of these two lines with all the rings up to N_m define a set of parallel segments, which constitute the LGS for the link ending at $Q_{(m)}$ (see Fig. 1*a*).
- parallel tolerance band: the decision cone from the center to midpoints $Q_{l,(m)}$ and $Q_{r,(m)}$ is replaced by a decision 'tube', with sides that are parallel to the link ending at $Q_{(m)}$ and pass through $Q_{l,(m)}$ and $Q_{r,(m)}$, respectively (see Fig. 1*b*). Intuitively, a parallel quantiser uses higher order rings more often than a triangular quantiser because the segments lying on the lower order rings are bigger. Accordingly, a parallel quantiser will result in a smaller number of links to be encoded, but will also result in a higher quantisation error.

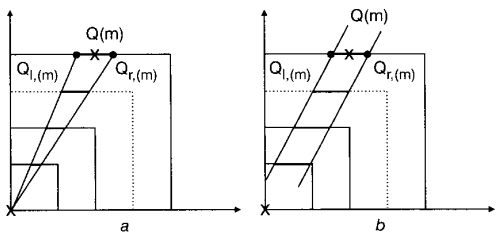

Fig. 1 *Link gate set associated with a node $Q_{(m)}$*
a Using triangular tolerance band
b Using parallel tolerance band

The quantisation procedure is then as follows: the pen trajectory is first broken into a sequence of 'traces' (i.e., a complete pen-down movement bounded by two pen-up movements); the first point of each trace is kept. For all others, the concentric rings are centred on the previous quantised point. For each ring, starting from the highest order one, we check if there is a LGS completely intersected by the trajectory curve. If not, we examine the next lower order ring. Otherwise, the link associated with the LGS defines the next quantisation point. To regenerate the quantised curve, one just has to draw straight lines between successive quantised points.

2.1 MRDCC encoding

After the quantisation step, each trace of the pen trajectory is represented as: $p_0 + \langle d_1, d_2, \ldots \rangle$, where d_i is a numeric label associated with the ith link. A common labelling convention is to number the links in a counter-clockwise manner from 0 to $\sum_{k=1}^{m} 8N_k - 1$. Therefore, with fixed-wordlength encoding, $\lceil \log_2(\sum_{k=1}^{m} 8N_k) \rceil$ bits are needed to represent each link. However, this requirement easily proves unnecessarily high for smooth curves when differential techniques are used.

One differential chain coding approach, the so-called DCC proposed by Arnbak and colleagues at Delft University [12–14], avoids variable-wordlength encoding with the following scheme: if the difference $\delta_i = d_i - d_{i-1}$ between two consecutive links is equal to $+1$, 0, or -1, a two-bit codeword is produced (relative encoding); otherwise a $(2 + \lceil \log_2(8N_k) \rceil)$-bits codeword is used (absolute encoding; the first two bits are a prefix distinguishing from relative mode).

MRDCC is an extension of DCC to the multi-ring case [7, 8]: if two successive links have been quantised by the same ring, and if the difference equals $+1$, 0, or -1, then a two-bit relative codeword is used. Otherwise, an absolute code word of length $(2 + \lceil \log_2(\sum_{k=1}^{m} 8N_k) \rceil)$ bits is used. Although MRDCC is not necessarily optimal from an information-theoretical point of view, it is attractive for transmission of ink messages in terms of data syntax, decoding, and transmission error control.

To completely characterise the encoding procedure, one must also specify how to identify the start and/or end of each trace of ink. One solution is to define a 'pen-up' codeword (with length dependent on the particular chain code used); another is to encode the number of links within the trace. In any case, the first point of each trace has to be encoded using its absolute coordinates. In the following, we will consider the number of bits required to encode this aspect of the ink-structure as a constant, whatever the chain code is, and our numerical results will not reflect it.

3 Evaluation method

The MRDCC algorithm was tested on three sets of ink images from eighty-two different writers. In the first set, Chinese writers were presented with open-ended scenarios that required composing a message. These messages were collected on a Fujitsu Stylistic tablet computer in a window of dimensions 160×240 pixels. Two examples of the scenarios are:

- 'Your friend just got a big promotion at work. Write a message to acknowledge his/her accomplishment.'
- 'Send a couple of ideas/sketches of new product logos to your boss.'

a b

Fig. 2 *Sample images of short ink messages containing handwritten text and graphics*
a Chinese messages composed in a 160 × 240 pixel window
b English messages composed in a 160 × 100 pixel window

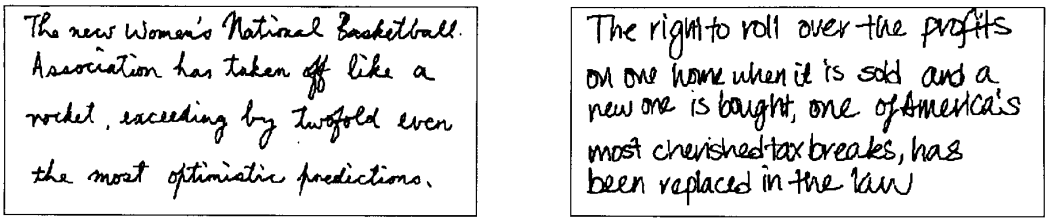

Fig. 3 *Sample images of long ink messages containing English handwritten text only*
Messages were composed inside a 500 × 250 pixel window

In the second set, English writers were presented with the same scenarios to elicit the messages using a window of 160 × 100 pixels. These window sizes were motivated by the form factor of actual PDA devices at our lab. Sample images from these two sets are shown in Fig. 2. These images were collected at screen resolution.

In the third set of images, writers were asked to transcribe a short paragraph from different news stories using a Wacom LCD tablet. Sample images from this set are shown in Fig. 3. These images were collected at tablet resolution, which is typically several times higher than screen resolution.

The details of our three datasets are summarised in Table 1.

3.1 Evaluation results

In this Section, we report on the evaluation of MRDCC in terms of coding efficiency and quantisation distortion. To compare the efficiency of different codes, we used actual relative compactness [6]. This is simply the ratio of the number of bits needed to encode some ink using a given multi-ring code and the number of bits needed when using a single ring only. Three MRDCC codes were evaluated: (1, 2)-MRDCC, (1, 3)-MRDCC and (1, 2, 3)-MRDCC. As described in Section 2, an $(N_1, \ldots, N_m)$-MRDCC code uses m concentric square rings in the quantisation stage and multi-ring DCC for encoding. Table 2 shows the actual relative compactness we obtained using these codes averaged over all images in each dataset. Also shown are the

Table 1: Datasets of ink messages used for evaluation of MRDCC algorithm

Data set name	Total images	Total writers	Graphics and text	Window size	Ink resolution
Chinese short messages (CSM)	120	20	both	160 × 240	low
English short messages (ESM)	115	20	both	160 × 100	low
English long messages (ELM)	84	42	text only	500 × 250	high

Table 2: Comparison of MRDCC and DCC codes in terms of actual relative compactness when using parallel tolerance band quantisation

Data set	(1)-DCC	(1, 2)-MRDCC	(1, 3)-MRDCC	(1, 2, 3)-MRDCC
CSM	1	1.06 (0.80, 1.31)	0.96 (0.58, 1.13)	1.14 (0.76, 1.36)
ESM	1	1.09 (0.76, 1.25)	1.00 (0.65, 1.15)	1.20 (0.82, 1.37)
ELM	1	1.01 (0.82, 1.16)	0.91 (0.70, 1.04)	1.02 (0.77, 1.16)

Maximum and minimum relative compactness values are also given in parentheses. Globally, (1, 3)-MRDCC is most efficient code

minimum and maximum actual relative compactness values. In all cases, the parallel tolerance band criteria was employed during quantisation.

For the majority of images in our three datasets, (1, 2)-MRDCC and (1, 2, 3)-MRDCC are less efficient than DCC. Specifically, in 83% of the CSM images, 88% of the ESM images and 61% of the ELM images, DCC is more efficient. These results contradict the geometrical analysis developed by Johannessen *et al.* [7, 8], where it was suggested that MRDCC always offers the highest coding efficiency. Moreover, we found that in over 76% of all images (1, 2, 3)-MRDCC does not improve over (1, 2)-MRDCC, also in contradiction to their theoretical results.

Although Johannessen's analysis is based on parallel tolerance band quantisation only, we also computed relative compactness for the triangular tolerance band case (see Table 3). We observe that in the three datasets, a single ring code almost always outperforms multi-ring codes (1, 2)-MRDCC and (1, 2, 3)-MRDCC. Compared to (1, 3)-MRDCC, DCC was better in over 94% of the images.

In order to compare the codes in terms of quantisation error, we used a per-length distortion measure given by the average value of a distortion function evaluated at each grid point of the original ink. Our function is defined as the Euclidean distance (measured in pixels) from the considered pixel of the curve to the closest pixel of the regenerated quantised curve. The per-length distortions, averaged over all images in each dataset, together with the maximum distortion measured on each dataset, are presented in Table 4. As expected, (1)-DCC is almost lossless. We also observe that (1, 2, 3)-MRDCC can sometimes offer an improvement over (1, 2)-MRDCC. Code (1, 3)-MRDCC is less lossy than (1, 2, 3)-MRDCC because when ring-2 is removed, sections of the curve that would have been quantised with ring-2 are now quantised using ring-1.

4 Relative vector probabilities in MRDCC and DCC

To state the superiority of MRDCC over DCC, Johannessen *et al.* used a theoretical model of the MRDCC per-length bit-rate. Their model relies strongly on two different parameters: PR_n, the probability of occurrence for ring n, and PV_n, the relative vector probability within ring n. The computation of these probabilities is based upon a theory concerning single ring differential chain coding [15]. Johannessen's model implicitly relies on a strong assumption: whatever the chain code is, PV_n remains the same as in the DCC case. However, in our datasets, we actually found that this assumption never holds. Instead, when averaged over all images in each tested dataset, we obtained $PV_n^{DCC} > PV_n^{MRDCC}$ (see Table 5). On an image by image analysis, this result was always true for the high-resolution ELM dataset. It was also true in all CSM and ESM cases when $n = 1, 2$, in 99% of the CSM cases when $n = 3$, and in over 80% of the ESM cases when $n = 3$.

We tried to characterise the relationship between PV_n^{MRDCC} and PV_n^{DCC}. Under the assumption that MRDCC behaviour can be locally approximated by DCC, consider a segmentation of the curve into smaller sections where only one ring is used. Let l_n be a random variable giving the number of successive vectors in such a curve section where only ring n is used. The number of relative vectors in this section of the curve can then be expressed as $(l_n - 1) \times PV_n^{DCC}$. Note that a -1 is needed

Table 3: Comparison of MRDCC and DCC codes in terms of actual relative compactness when using triangular tolerance band quantisation

Data set	(1)-DCC	(1, 2)-MRDCC	(1, 3)-MRDCC	(1, 2, 3)-MRDCC
CSM	1	1.27 (0.96, 1.46)	1.14 (0.89, 1.27)	1.38 (1.01, 1.55)
ESM	1	1.30 (1.05, 1.43)	1.17 (0.88, 1.28)	1.40 (1.04, 1.59)
ELM	1	1.32 (1.03, 1.47)	1.11 (0.86, 1.22)	1.52 (1.10, 1.73)

Maximum and minimum relative compactness values are also given in parentheses.

Table 4: Comparison of MRDCC and DCC codes in terms of per-length distortion

Data set	(1)-DCC	(1, 2)-MRDCC	(1, 3)-MRDCC	(1, 2, 3)-MRDCC
CSM	0.02 (1.00)	0.10 (1.41)	0.06 (1.41)	0.09 (1.41)
ESM	0.01 (1.00)	0.12 (1.41)	0.05 (1.41)	0.09 (1.41)
ELM	0.01 (1.00)	0.08 (1.41)	0.09 (2.00)	0.09 (2.24)

Maximum distortion is also included in parentheses

Table 5: Actual relative vector probabilities (PV_n) for different codes averaged over each tested dataset

Code	(1)	(2)	(3)	(1, 2)		(1, 3)		(1, 2, 3)		
Data set	PV_1	PV_2	PV_3	PV_1	PV_2	PV_1	PV_3	PV_1	PV_2	PV_3
CSM	0.87	0.76	0.63	0.06	0.63	0.31	0.51	0.06	0.07	0.51
ESM	0.80	0.62	0.43	0.07	0.52	0.34	0.38	0.08	0.05	0.38
ELM	0.99	0.97	0.94	0.06	0.78	0.22	0.64	0.05	0.03	0.64

Table 6: Actual relative vector probabilities and their estimated values on ELM dataset

Chain code	(1)	(2)	(3)	(1, 2)		(1, 3)		(1, 2, 3)		
Probability	PV_1	PV_2	PV_3	PV_1	PV_2	PV_1	PV_3	PV_1	PV_2	PV_3
Actual value	0.99	0.97	0.94	0.06	0.78	0.22	0.64	0.05	0.03	0.64
Estimation (using L_n)	-	-	-	0.06	0.77	0.22	0.63	0.05	0.09	0.62
Error	-	-	-	0.00	0.01	0.00	0.01	0.00	0.06	0.02

Table 7: Comparison of theoretical and actual relative compactness of MRDCC codes

Code	(1, 2)-MRDCC				(1, 3)-MRDCC				(1, 2, 3)-MRDCC			
Data set	PV_n^{MRDCC}		PV_n^{DCC}		PV_n^{MRDCC}		PV_n^{DCC}		PV_n^{MRDCC}		PV_n^{DCC}	
CSM	1.08	(0.02)	0.73	(0.33)	0.98	(0.02)	0.66	(0.30)	1.15	(0.01)	0.68	(0.46)
ESM	1.12	(0.03)	0.81	(0.28)	1.02	(0.02)	0.76	(0.24)	1.22	(0.02)	0.83	(0.37)
ELM	1.02	(0.01)	0.58	(0.43)	0.91	(0.00)	0.46	(0.45)	1.02	(0.00)	0.45	(0.57)

Theoretical values are computed using PV_n^{MRDCC} and PV_n^{DCC}. Absolute deviation from actual values is shown in parentheses

because the first vector is always an absolute vector. Therefore, we can write the following approximation to PV_n^{MRDCC}:

$$\hat{PV}_n^{MRDCC} = \frac{L_n - 1}{L_n} \times PV_n^{DCC} \text{ where } L_n = E[l_n] \quad (1)$$

This approximation is consistent with our empirical result that globally PV_n^{DCC} is greater than PV_n^{MRDCC}. In order to validate the accuracy of this approximation, we computed the average value of l_n for each of the multi-ring codes we have been examining. Table 6 shows the actual relative vector probabilities and their estimated values on the high-resolution ELM dataset. A very small error in the approximation is observed overall. In the case of $PV_2^{(1,2,3)}$, where the error is higher, we had very few samples to estimate L_2 because the ring probability $PR_2^{(1,2,3)}$ was only 0.06. When we tested our approximation on the lower-resolution datasets, a larger overall error was observed due to the same kind of estimation difficulties.

We now use PV_n^{MRDCC} to express the number of bits required to encode one vector taken from ring-n in a (K)-*MRDCC* code as

$$r_n^{(K)-MRDCC} = 2 \times PV_n^{(K)-MRDCC} + \left(2 + \left\lceil \log_2 \left(\sum_{i \in (K)} 8i \right) \right\rceil\right) \times (1 - PV_n^{(K)-MRDCC}) \quad (2)$$

and then extend to the MRDCC case the original theoretical formula given by Freeman [6], to estimate the relative compactness:

$$\frac{B^{(K)}}{B^{(1)}} = \frac{\sum_{i \in (K)} PR_i \times r_i^{(K)-MRDCC}}{\left(\sum_{i \in (K)} (i \times PR_i) \times (2 \times PV_1^{DCC} + 5 \times (1 - PV_1^{DCC}) \right)} \quad (3)$$

where $B^{(K)}$ is the number of bits needed to encode the curve with $(K) - MRDCC$, and PR_i is the probability of ring i. Note that, as in the original case, this expression includes the assumption that lengthwise using a vector from ring n is equivalent to using n vectors from ring 1.

To further underline our point that one cannot accurately model the behaviour of MRDCC by assuming that PV_n^{MRDCC} can be replaced by PV_n^{DCC}, we use eqn. 3 to estimate relative compactness with both values, and compare the result with the actual relative compactness reported in Table 2. The two estimates for each data set, together with their absolute deviation from the actual relative compactness, are shown in Table 7.

We observe in the Table that there is a significant discrepancy between actual and estimated relative compactness when Johannessen's assumption is used.

5 Choosing the right code

Although eqn. 3 could be used to find the best multi-ring chain code for each of our datasets, we thought of a more direct approach based upon the guidelines suggested by Freeman [6]. The procedure involves computing and analysing the probability distribution for all rings up to a maximum.

In Fig. 4, we present the probability distribution of the rings up to the 10th order for the CSM and ESM datasets, computed using a grid size of $\tau = 1$ pixel (the smallest possible value). Because of the shape and position of the envelope of the distribution, we can conclude that this grid size is appropriate. Based on the peaks in the envelope, it appears that for these two datasets rings 1 to 4 are the most important ones.

Accordingly, we now examine the distribution limited to rings 1, 2, 3 and 4 only (see Fig. 5). It is evident that by removing the higher order rings, we have added that probability mass to the lower order rings, and thus their

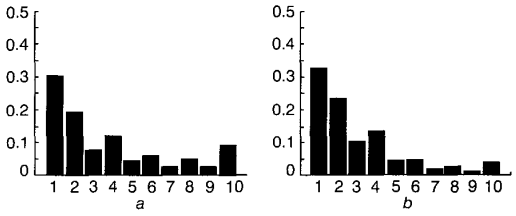

Fig. 4 *Ring probabilities PR_i, $1 \le i \le 10$, computed using a (1, 2, ..., 10)-GCC quantiser and a grid size of $\tau = 1$ pixel*

a CSM dataset
b ESM dataset

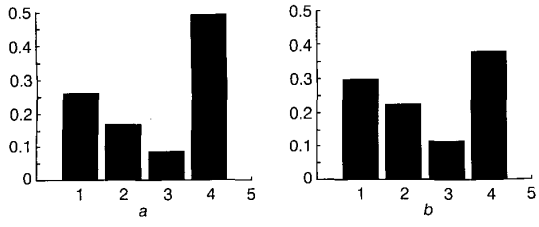

Fig. 5 *Ring probabilities PR_i, $1 \leq i \leq 4$, computed using a (1, 2, ..., 4)-GCC quantiser and a grid size of $\tau = 1$ pixel*

a CSM dataset
b ESM dataset

relative vector probabilities will increase as well. Ring 3 has the lowest probability. Because low-probability rings are mostly used for the encoding of absolute vectors, and their inclusion in a code increases the number of bits per absolute vector, we concluded that only (1, 2, 4)-MRDCC and (1, 4)-MRDCC were worth testing.

For the CSM dataset, we found the best code to be (1, 4)-MRDCC, with a relative compactness of 0.93 and a per length distortion of 0.13 pixels. For the ESM dataset, (1, 4)-MRDCC was also the best code, with a relative compactness of 0.98 and a per length distortion of 0.12 pixels. Intuitively, a per-length distortion of 0.1 pixels means that, on average, 1 pixel out of 10 is shifted by one from its original position. We believe this is almost imperceptible in a 160×240 image, and thus quite acceptable in our intended application.

Choosing a chain code for the ELM dataset is a little bit more complicated because of its high resolution. The envelope of the probability distribution computed using grid size $\tau = 1$ is quite flat, and with the most significant peak located far to the right (see Fig. 6*a*). This is a clear indication that the grid size is too fine. Indeed, a coarser grid size of $\tau = 20$ pixels results in a more desirable distribution envelope (see Fig. 6*b*). With this grid size, however, the precision will be low.

In order to keep a higher precision, we recomputed the distributions using $\tau = 4$ (see Fig. 7). There are now two

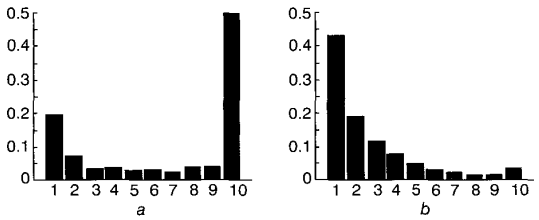

Fig. 6 *Ring probabilities PR_i, $1 \leq i \leq 10$, for the ELM dataset computed using a (1, 2, ..., 10)-GCC quantiser*

a Grid size $\tau = 1$ pixel
b Grid size $\tau = 20$ pixels

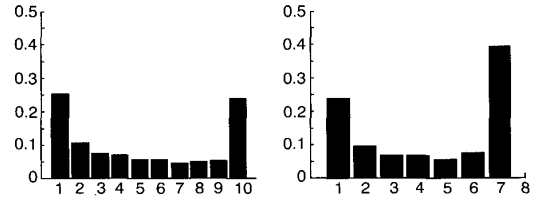

Fig. 7 *Ring probabilities for the ELM dataset computed using a grid size of $\tau = 4$ pixels*

a PR_i, $1 \leq i \leq 10$, computed with a (1, 2, ..., 10)-GCC quantiser
b PR_i, $1 \leq i \leq 7$, computed with a (1, 2, ..., 7)-GCC quantiser

sharp peaks, of about the same size, at the far right and far left of the distribution. As mentioned above, the peak to the right represents the sum of higher-order ring probabilities. It thus seems clear that we have to choose a two-ring chain code, i.e., (1, n)-MRDCC for some n close to 10. A good tradeoff for choosing n is to make the expression $\log_2(8 \times (1 + n))$ equal to or slightly lower than an integer; not too big (low ring and relative vector probabilities), and not too small (high relative vector probability, but too many small steps). Among integers in the interval [7, 15] of values close to 10 that meet these criteria, 7 would seem to be the most appropriate choice for n. Using (1, 7)-MRDCC we obtained a relative compactness of 0.78 and a per-length distortion of 0.84 pixels. We empirically confirmed the superiority of this code by trying many other ring combinations. The higher per-length distortion, compared to the short messages dataset, has to be interpreted in the context of a 5000×2500 image size, i.e., 500×250 window size at a 1000 d.p.i. tablet resolution.

6 Conclusions

Two-way transmission of electronic ink, particularly if wireless, offers users of personal digital assistants and Internet appliances a compelling new way to communicate. With this application domain in mind, we evaluated the performance of single-ring and multi-ring differential chain coding algorithms on a large number of images. Multi-ring differential chain coding (MRDCC) is not necessarily optimal from an information-theoretical point of view; however, it is attractive for transmission of ink messages in terms of data syntax, decoding, and transmission error control. We established that the coding efficiency of (1, 2)-MRDCC and (1, 2, 3)-MRDCC is not always better when compared to (1)-DCC as previously suggested. Specifically, DCC was more efficient than (1, 2)-MRDCC and (1, 2, 3)-MRDCC in over 80% of our low-resolution test images and in over 61% of our high-resolution test images. We have also shown that for a given ring, the probability of relative vectors in the multi-ring case is almost always lower than in the single-ring case. As a consequence of this, one cannot accurately model the behaviour of MRDCC based on PV_n estimates from the DCC case. Another consequence is that the improvements in efficiency that can be obtained by using differential techniques are higher for the single-ring codes and lower for the multi-ring ones. As our experimental results have shown, choosing the right chain code and grid size also depends on the resolution and geometrical properties of the images to be encoded. Fortunately, as we have illustrated, Freeman's guidelines involving an analysis of the ring probability spectrum can be effectively used to find a code that meets the constraints of the target application. Using this method, we found (1, 4)-MRDCC to be the most efficient code for our short messages datasets, which were composed in a PDA-size window at screen resolution. This multi-ring code resulted in a very small per-length distortion and an average message size of 385 bytes for English text and 537 bytes for Chinese text. These values represent a 71% efficiency relative to the uncompress average message size and an 18% efficiency relative to a lossless scheme based on Huffman coding. To put these compressed message sizes in context, today's popular SMS text messages on cellular networks are commonly restricted to a maximum of about 640 bytes. On the larger message dataset, collected at high resolution, the best multi-ring code resulted in an average message size of

13.3 kbytes, or a 79% efficiency relative to the uncompressed average message size and a 36% efficiency relative to a lossless scheme based on Huffman coding.

7 References

1 ITU-T Recommendation T.150: 'Telewriting terminal equipment'. International Telecommunication Union, 1993
2 FREEMAN, H.: 'Computer processing of line-drawing data', *Comput. Surv.*, 1974, **6**, pp. 57–96
3 BONS, J.H., and KEGEL, A.: 'On the digital processing and transmission of handwriting and sketching'. Proceedings of EUROCON 77, pp. 2.4.6.1–6, Venice, 1977
4 FREEMAN, H.: 'Application of the generalized chain coding to map data processing'. Proceedings of IEEE computer society conference on *Pattern recognition and image processing*, Chicago, IL, 1978, pp. 220–226
5 WILSON, S.G., and SCHOLTEN, D.K.: 'Chain coding with a hexagonal lattice', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1983, **5**, (5), pp. 526–533
6 SAGHRI, A., and FREEMAN, H.: 'Generalized chain codes for planar curves'. Proceedings of 4th international joint conference on *Pattern recognition*, 1978, pp. 701–703
7 JOHANNESSEN, A.B., BONS, J.H., WEYLAND, N.B.J., and PRASAD, R.: 'Multiring differential chain codes for line drawings', *Electron. Lett.*, 1990, **26**, (10), pp. 677–678
8 JOHANNESSEN, A.B., PRASAD, R., WEYLAND, N.B.J., and BONS, J.H.: 'Coding efficiency of multiring differential chain coding', *IEE Proc. I, Commun. Speech Vis.*, 1992, **139**, (2), pp. 224–232
9 BONS, J.H., and PRASAD, R.: 'Enhanced differential chain coding for transmission of high quality line graphics', *Electron. Lett.*, 1994, **30**, (10), pp. 768–769
10 CHUNG, J.W., MOON, J.H., and KIM, J.K.: 'Conditional differential chain coding for lossless representation of object contour', *Electron. Lett.*, 1998, **34**, (1), pp. 55–56
11 SAGHRI, A., and FREEMAN, H.: 'Comparative analysis of line drawing modeling schemes', *Comput. Graph. Image Process*, 1980, **12**, pp. 203–223
12 ARNBAK, J.C., BONS, J.H., and VIEVEEN, J.W.: 'Graphical correspondence in electronic-mail networks using personal computers', *IEEE J. Sel. Areas Commun.*, 1989, **7**, (2), pp. 257–267
13 DE KLERK, M., PRASAD, R., BONS, J.H., and WEYLAND, N.B.J.: 'Introducing high-resolution line graphics in UK teletext using differential chain coding', *IEE Proc. I, Commun. Speech Vis.*, 1990, **137**, (6), pp. 325–334
14 PRASAD, R., and LIU, K.: 'On the quantization distorsion and coding efficiency in line drawing transmission using differential chain coding'. Proceedings of IEEE international conference Communications 90, 1990, **3**, pp. 325A.2.1–325A.2.5
15 PRASAD, R., VIEVEEN, J.W., BONS, J.H., and ARNBAK, J.C.: 'Relative vector probabilities in differential chain-coded line drawings'. Proceedings of IEEE Pacific rim conference on *Communications, computers and signal processing*, Canada, 1989, pp. 138–142
16 FREEMAN, H.: 'The encoding of arbitrary geometric configuration', *IRE Trans. Electron. Comput.*, 1961, **EC-10**, pp. 260–268
17 SAGHRI, A., and FREEMAN, H.: 'Analysis of the precision of generalized chain codes for the representation of planar curves', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1981, **PAMI-3**, (5), pp. 533–539
18 LIU, K.: 'Model for line-drawing graphics and differential angle distribution', *Electron. Lett.*, 1989, **25**, (22), pp. 1473–1474
19 KOPLOWITZ, J.: 'On the performance of chain codes for quantization of line drawings', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1981, **PAMI-3**, (2), pp. 180–185