MVA'90 IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

DECISION FEEDBACK EQUALISATION FOR DOCUMENT INPUT

John Bernsen

Philips Research Laboratories, WB-2,
5600 JA Eindhoven, The Netherlands.

Abstract

Obtaining digital images from binary scenes (for example text on paper) and subsequent thresholding does not necessarily lead to high quality binary images. In the imaging process, the binary scene is low-pass filtered and tiny details are therefore smeared. This low-pass filtering is necessary for the subsequent sampling to get a discrete image. In this paper, we borrow from the theory of data transmission, where a similar smearing process occurs: inter symbol interference (ISI) [Qu85]. We have tried to use the theory of decision feedback equalisation (DFE) in order to improve the quality of detail of binary images from binary scenes. It will be shown that DFE can be used to obtain a binary image with the same quality of a surface which is 2 to 3 times as large as when straightforward thresholding is used. The method will be compared with dynamic thresholding [Be86] and iterative constrained deblurring [BL90]. Another application of DFE in images is in image enhancement.

Introduction

Thresholding of an image obtained with some imaging system from a binary scene (e.g. text on paper), does not necessarily result in a binary image which is a faithful representation of that binary scene. First of all, this is due to the sampling density, but there is another reason as well. The compulsory anti-aliasing low-pass filter causes some blur. In a grey-level image from a binary scene like printed text, where the pixel size is just a little bit smaller than the holes in characters, these holes (in for example the characters "e" and "a") might look white, but they may be just a little less black than the ink around these holes. This is due to the low-pass filtering. With a threshold at the 50% level, these holes may be thresholded to black.

One solution would be to increase the sampling density and to increase the cut-off frequency of the low-pass filter accordingly, but then we need larger cameras, more scans, etc. Besides, it does not seem to be necessary, because in the grey-level image, small details such as holes in characters are still visible.

Another solution comes from the field of data-transmission. In this field, data are transmitted over a channel e g. using binary pulses. Usually, one wants to have the highest possible data-rate over a given channel. The channel tends to spread these pulses, so that they will overlap at high data-rates: Inter Symbol Interference (ISI) [Qu85]. It can be overcome using decision feedback equalisation (DFE). The principle involved is that the receiver takes a (good!) guess at what a transmitted pulse might have been. Using the outcome of this guess, the receiver computes what the ISI will be due to this pulse and then subtracts this estimated ISI from the received signal, so that the ISI is effectively removed. Succeeding pulses can then be detected in a signal that does not contain ISI from previous pulses anymore. It seems that it is possible now to transmit a signal with a large bandwidth over a channel with a small bandwidth. However, what happens is that *bandwidth* is traded for *dynamic range.* This is only possible if the signal to noise ratio (SNR) is high enough.

The blurring of images from binary scenes can be modeled as 2-d ISI and DFE can be generalised to remove it. The most difficult problem is that in a binary scene, the data (e.g. ink on paper) can start and stop at any position (e.g. halfway a pixel), because it is a *continuous* scene. The DFE theory is derived for a system where the original signal is *discrete* and one knows the data rate as well as the sampling phase [Qu85] and [MS81]. A simple adaptation gives already quite good results as will be explained in Section 2. We will show that DFE is in effect an adaptive thresholding method similar to dynamic thresholding [Be86]. The difference between the two is that the former uses more knowledge over the image formation process and assumes that the original unblurred image is binary. The latter technique tries to make a binary image of a grey-level one without using any assumptions about the input image. We will show that the use of this extra knowledge by DFE yields a better result.

In Section 1 we treat 1-d DFE. In Section 2, we extend DFE to the 2-d case. In Section 3, we discuss some experiments where we use a *discrete* original unblurred image and compare DFE to iterative constrained deblurring [BL90]. Also in Section 3, we deblur images taken in with a *camera* from a binary scene and compare DFE with dynamic thresholding [Be86] for this situation. Finally, in Section 4, we draw some conclusions.

1 DFE: the 1-d case

Decision feedback equalisation (DFE) in itself or DFE using an extra receiver filter can be used to equalise channels which exhibit inter-symbol interference (ISI). Without equalisation, the detection of transmitted symbols can be difficult or even impossible for such channels. For an overview article on equalisation methods, refer to [Qu85]. DFE uses the already detected symbols to estimate the ISI they cause in order to make the detection of future symbols easier. In this section, we will explain DFE and how filters for DFE can be designed. Most of the material covered in 1.2 originates from [Qu85] and [MS81].

1.1 DFE of causal channels

Suppose a signal a_i with signal levels +1 and -1 is transmitted over a channel with impulse response p_i, for which the following applies:

$$(1)\quad p_i \;=\; 0 \quad ; i<0 \text{ (causal channel),}$$
$$\qquad\;\; = \; 1 \quad ; i=0 \text{ (amplitude scaling).}$$

This channel can be equalised using DFE as shown in Fig. 1 (without the receiver filter which will be discussed in Section 1.2). We will show that the equalisation can be done perfectly if the noise level is sufficiently low.

In the sequel, $\otimes$ denotes convolution and $(a \otimes b)_i$ denotes element i of the result of the convolution of a and b. The signals in the system in Fig. 1 are computed as follows. The received signal r_i is the output of the channel corrupted by additive white Gaussian noise v_i:

$$(2) \quad r_i \quad = \quad (a \otimes p)_i \; + \; v_i \quad .$$

The output signal $\hat{a}_i$ is the output of the quantiser Q with the decision signal $\bar{a}_i$ as input:

$$(3) \quad \hat{a}_i \; = \; \mathrm{Q}(\bar{a}_i) \; = \; 1 \text{ if } \bar{a}_i > 0 \; ; \; \hat{a}_i \; = \; -1 \text{ if } \bar{a}_i \leq 0 \; .$$

From the output signal, the feedback filter estimates the ISI which would be caused by the transmission of $\hat{a}_i$, using the impulse response c_i. This estimated ISI signal is subtracted from the received signal r_i (no receiver filter in this case) to form the decision signal $\bar{a}_i$:

$$(4) \quad \bar{a}_i \quad = \quad r_i \; - \; (\hat{a} \otimes c)_i \quad .$$

The feedback filter has access to *past* output elements only, so the coefficients c_i have to be zero for i<0. Assuming that the quantiser makes the right decisions, $\hat{a}_i$ will be equal to a_i for all i. Taking this into account, $\bar{a}_i$ becomes:

$$(5) \quad \bar{a}_i \quad = \quad (a \otimes p)_i \; + \; v_i \; - \; (a \otimes c)_i \quad .$$

If we choose the feedback filter coefficients such that:

$$(6) \quad c_i \quad = \quad 0 \;\text{ for }\; i = 0 \;\; ; \;\; c_i \;\; = \;\; p_i \;\text{ for }\; i \neq 0 \quad ,$$

we get:

$$(7) \quad \bar{a}_i \quad = \quad a_i \, p_0 \; + \; v_i \;\; = \;\; a_i \; + \; v_i \quad .$$

So whenever $|v_i| < 1$, $\bar{a}_i$ can be used to estimate a_i correctly, whereby our assumption of the correctness of $\hat{a}_i$ is met. Because we assumed that the channel is causal, it is possible to use the feedback filter coefficients according to Eq. (6).

From this, one can get the impression that any data rate is possible over narrow bandwidth channels. However, bandwidth is exchanged with dynamic range. Disregarding the noise, the peak value of r is:

$$(8) \quad r_{\text{peak}} = \sum_j |p_{i-j} \, a_j| \;\; = \;\; \sum_j |p_j| \geq 1 \quad ,$$

where we also used Eq. (1). The peak value of the signal a_i we are interested in is only 1. The peak value of the noise should be less than 1 in order to estimate a_i correctly. Suppose we have a certain channel with a low-pass filter characteristic. If we lower the cut-off frequency, while rescaling the coefficients of the new impulse response according to Eq. (1), then r_{peak} will increase. Therefore, we must have a better SNR at the output of the channel in order to be able to retrieve a_i correctly when the cut-off frequency is lowered.

1.2 DFE in combination with a FIR receiver filter

In Section 1.1 we saw the importance of the relative magnitude of p_0 of the channel impulse response with respect to the other samples. We can shift the channel impulse response backwards in time to have another non-zero sample as p_0, possibly resulting in a smaller r_{peak} in Eq. (8) and a lower required SNR. However, the resulting channel will be non-causal and cannot be equalised totally using only DFE. We introduce the receiver filter (Fig. 1) as a more general method to shift the channel impulse response in time. This filter is designed such that at the detector, the noise and ISI levels become similar in order to have the lowest error signal level possible. If the SNR at the receiver filter input is high enough, the receiver filter will be designed such that the noise level is increased, while decreasing the ISI that cannot be equalised by the DFE. This is the result of a least squares optimisation of the "error" $\bar{a}_i$ - a_i [Qu85]. Now, we have no restrictions on the channel p_i. It can be non-causal. Note that much of the theory of the preceding section still applies, by considering the cascade of the channel and the receiver filter as the "total channel" which is to be equalised by DFE. The set W contains all the indices for which the receiver filter has a non-zero coefficient. The receiver filter can be non-causal. The feedback filter has to be causal of course, because future decisions cannot be used. So:

$$(9) \quad c_i \quad \begin{aligned} &\neq 0 \, ; i \in C \; (C \text{ is a set of } \textit{positive} \text{ indexes}), \\ &= 0 \, ; \text{elsewhere.} \end{aligned}$$

Using again the assumption that the decisions made in the quantiser are correct (i.e. $\hat{a}_i = a_i$), we get for $\bar{a}_i$:

$$(10) \quad \begin{aligned} \bar{a}_i \;&=\; b_i \; - \; (a \otimes c)_i \;\; = \;\; (w \otimes r)_i \; - \; (a \otimes c)_i \\ &= \;\; (w \otimes v)_i \;\; + \;\; (w \otimes p \otimes a)_i \;\; - \;\; (a \otimes c)_i \quad . \end{aligned}$$

This system can be optimised by minimising the mean squared difference between a_i and $\bar{a}_i$, through adjusting w_i and c_i. The error e to be minimised is (*assuming uncorrelated zero mean noise and data*, with variances $\sigma_v{}^2$ and $\sigma_a{}^2$, respectively):

$$(11) \quad \begin{aligned} e \;\; &= \;\; \mathrm{E}\{(\bar{a}_i \; - \; a_i)^2\} \\ &= \sigma_v{}^2 \sum_i | \, w_i|^2 + \sigma_a{}^2 \sum_i \, [(w \otimes p)_i \; - \; c_i \; - \; \delta_i \,]^2 . \end{aligned}$$

Minimising e with respect to c_k for all $k \in C$ is achieved by taking the partial derivatives of e with respect to c_k and setting the results to zero. This yields:

$$(12) \quad \begin{aligned} \frac{\partial e}{\partial c_k} \;\; &= \;\; \sigma_a{}^2 \cdot \sum_{i=-\infty}^{\infty} \Big\{ 2 \, [\, (w \otimes p)_i \; - \; c_i \; - \; \delta_i \,] \cdot \\ &\qquad \cdot \frac{\partial}{\partial c_k} [\, (w \otimes p)_i \; - \; c_i \; - \; \delta_i \,] \Big\} \\ &= \;\; -2 \, \sigma_a{}^2 \, [(w \otimes p)_k - c_k - \delta_k \,] \;\; , \; \forall \, k \in C \quad . \end{aligned}$$

Setting (12) to zero yields:

$$(13) \quad c_k \quad = \quad (w \otimes p)_k \quad , \quad \forall \, k \in C \quad .$$

This is no surprise, since now the feedback filter has to equalise the "total channel" formed by the cascade of channel and receiver filter, which indeed has the convolution of both impulse responses as its impulse response.

Minimising e with respect to all the elements of the receiver filter w_k, with $k \in W$ means again taking partial derivatives and setting the results to zero. Using (13) as well yields:

$$(14)\quad \frac{\partial e}{\partial w_k} = 2\,\sigma_v{}^2\, w_k + \sigma_a{}^2 \cdot \sum_{i=-\infty}^{\infty} \Big\{ 2\,[(w \otimes p)_i - c_i - \delta_i] \cdot \cdot \frac{\partial}{\partial w_k} [(w \otimes p)_i - c_i - \delta_i] \Big\} \quad , \ \forall\, k \in W \quad .$$

The right-most term leads to:

$$(15)\quad \frac{\partial}{\partial w_k} [\,(w \otimes p)_i - c_i - \delta_i\,] \;=\; \frac{\partial}{\partial w_k} (w \otimes p)_i \;=\; p_{i-k} \ , \ \forall\, k \in W \quad .$$

Substituting this in Eq. (14) and setting (14) to zero yields:

$$(16)\quad \sum_{l \in W} \Big(w_l \sum_{i \notin C} p_{i-l}\, p_{i-k} \Big) + (\sigma_v{}^2 / \sigma_a{}^2)\, w_k \;=\; p_{-k} \ , \quad \forall\, k \in W \quad .$$

Eq. (16) is a system of k linear equations with k unknowns w_{k} and can be solved numerically if the channel impulse response, the noise variance and the data variance are known.

So far, we assumed that the output data $\hat{a}_{\mathrm{i}}$ are equal to the transmitted data a_{i}. If an error occurs, the feedback filter will be filled with one wrong datum and hence, instead of estimating the right amount of ISI to be subtracted from the "channel" output, it estimates something wrong which can make matters worse than when no equalisation was performed. For this reason, one single error can cause error propagation. This error propagation leads to an error rate which can be a factor of 10 to 100 worse than the error rate which can be computed using the actual SNR and assuming the noise to be Gaussian [Qu85]. Since error rates decrease easily by a factor of 10 for every dB in SNR improvement for an SNR around 15 dB [BP79], using an SNR which is a few dB better than required for systems without error propagation compensates for this effect.

2 DFE: the 2-d case

When considering an image of a binary scene (e.g. printed text), we have the same effect as what happens in data transmission. The effect of the camera-blur can be compensated for using DFE and a receiver filter, *if we use extra prior information* about the camera characteristics, noise level and black/white intensities of ink and paper. The black and white on the paper are the symbols that are to be transmitted over the 2-d channel which is formed by the camera and sampling devices. The sampled and digitised image is the output of the 2-d channel. We get a system like the one shown in Fig. 1, but all signals are 2-d now.

Note that the output signal of the feedback filter acts as a threshold. If we use a receiver filter, then not the "received" input image, but its enhanced version is thresholded with it. So DFE (with or without receiver filter) is in effect an algorithm which tries to compute an optimal threshold for each pixel, just like dynamic thresholding [Be86]. Dynamic thresholding is a technique which tries to estimate a threshold for the middle pixel in a window, using *no prior knowledge* and using only the information from the window.

The following problems make it difficult to use the theory of the preceding section straightforwardly:

1. We are dealing with a *two dimensional* channel now.
2. The *amplitudes* of the data are unknown.
3. We are dealing with *bounded* images.
4. The "*data-rate*" is unknown.

These problems will be treated in the following sections.

2.1 The mathematics of 2-d DFE

In Section 1.2 we discussed the mathematics of 1-d equalisation. Two linear filters are required, a non-causal receiver filter and a causal feedback filter. The receiver filter will not give any problems, because in image processing, it is quite natural to use linear non-causal filters. The set W contains all 2-d indices for which the receiver filter has a non-zero coefficient. The causal feedback filter will give a problem, however, because there is no natural ordering of past and future in an image. We have to make a choice of which pixels we assume to be the past ones and which should be the future ones. The set C contains all the non-zero coefficients of the feedback filter. Because of the required causality, we have the following constraint:

$$(17)\quad c_{m,n} \;=\; 0 \quad ; \quad m{<}0 \vee [\,(m{=}0) \wedge (n{\leq}0)\,] \quad .$$

Based on this, the mathematics for the 2-d case are merely a repetition of the mathematics of Section 1.2 with appropriate adaptations for 2-d summations. Therefore, we will just state the end results of the optimisation:

$$(18)\quad c_{k,l} \;=\; (w \otimes p)_{k,l} \ , \ \forall\, (k,l) \in C \quad .$$

$$(19)\quad \sum_{(r,s) \in W} \Big(w_{r,s} \sum_{(m,n) \notin C} p_{m-r,n-s}\, p_{m-k,n-l} \Big) + (\sigma_v{}^2 / \sigma_a{}^2)\, w_{k,l} \;=\; p_{-k,-l} \ , \ \forall\, (k,l) \in W \quad .$$

Eq. (19) is a system of linear equations with the same number of unknowns $w_{\mathrm{k,l}}$ and can be solved numerically if the channel impulse response, the noise variance and the signal variance are known. This seems rather cumbersome, but it only needs to be done once for a fixed camera and optical system.

2.2 The amplitudes of the 2-d data

In the original data transmission case, the amplitudes of the transmitted symbols are known. In our treatment of the 1-d and 2-d case, we assumed that the symbols are binary and that the data signal has no D.C. component. The last assumption was used during the computation of the expectancies of the signal energies. However, the binary scenes that we are looking at with our camera do not behave in this way. First of all, intensities are all positive. Secondly, both ink and paper can have any intensity. A solution to the second problem is that we use a calibrated or corrected set-up, such that the intensity coming from the paper is more or less constant all over the image and known, as well as the intensity from the ink. This only has to be done once for a fixed set-up which has its own lighting system. If this is done, the average of the paper intensity and the ink intensity is subtracted from all of the input image in order to solve the first problem. This works very good in practice.

2.3 Image boundary treatment

Images are finite in size and therefore the borders need careful treatment. We need previous output symbols in the DFE and these previous output symbols should be correct, because else the estimated ISI is very wrong. The previous output symbols needed at boundaries are at the left, upper and right boundary of the image. There is a difference in the treatment of each of the three.

At the right boundary, the output pixel $\hat{a}_{\mathrm{k,l}}$ that lies on the right boundary is copied to pixels on the same line outside the image to the right, to be used as previously estimated pixels in the processing of the subsequent lines.

For the upper boundary, there are no previous output pixels to copy. It is very difficult to get good estimates for the output pixels above the image. Here, we just take the upper line from the *output of the channel* and feed it through the quantiser without any processing. Its outputs are used as previous output pixels $\hat{a}_{k,l}$ above the image, where $\hat{a}_{k,l} = \hat{a}_{-1,l}$ for $k < -1$.

For *previous lines* at the left boundary, the previous output pixels from the left border column are copied. We still need the previous output pixels from *the same line to the left* outside the image. Assuming that pixels outside images are copies from border pixels, we can compute the required previous output pixels to the left outside the image, where H is the height of the image:

$$(20)\quad \bar{a}_{i,j} = b_{i,j} - (\hat{a} \otimes c)_{i,j} = \hat{a}_{i,j} = \hat{a}_{i,0} = \bar{a}_{i,0}\ ,$$
$$\text{for } (0 \le i < H) \wedge (j < 0) \quad \Rightarrow$$

$$\bar{a}_{i,j} = b_{i,0} - \sum_{m=1}^{\infty} \sum_{n=-\infty}^{\infty} (\hat{a}_{i-m,0-n}\, c_{m,n}) - \hat{a}_{i,0} \sum_{n=1}^{\infty} c_{0,n} \quad \Leftrightarrow$$

$$\bar{a}_{i,j} = [b_{i,0} - \sum_{m=1}^{\infty} \sum_{n=-\infty}^{\infty} (\hat{a}_{i-m,-n}\, c_{m,n})] \,/\, [1 + \sum_{n=1}^{\infty} (c_{0,n})],$$
$$\text{for } (0 \le i < H) \wedge (j < 0) \quad .$$

From $\bar{a}_{i,j}$, the quantiser Q computes the required previous output pixels at the left border in the same line.

2.4 The effect of the "unknown data rate"

In data transmission, the data symbols are transmitted with a fixed rate. In this paper, we are dealing with images of binary scenes. In these situations, there is no fixed data rate. The boundaries of the ink can be anywhere. To obtain pixels, the system to process the camera images has a certain horizontal and vertical sampling rate. It is possible that ink boundaries cut through a pixel, so that each of the two decisions: "pixel is black" or "pixel is white" is wrong.

A very simple solution, which at first does not seem consistent with the idea of decision feedback equalisation but which proved to work fine, is that instead of making binary decisions, we allow the quantiser to have more levels of output, say 20 or so. A pixel cut by an ink boundary is then estimated to be mid-grey. This multiple-level signal must be applied to the feedback filter. The output to be used for further processing can remain a multiple-level signal, but it can be thresholded as well.

3 Experiments

3.1 Deblurring of a discrete image

In [BL90], Biemond, Lagendijk and Mersereau describe iterative methods to deblur images using knowledge of the point-spread function of the blurring process. They show that the results can be improved if they use extra knowledge about the original input image, e.g. that it is binary. In one of the described experiments in [BL90], the filter used for blurring has a point-spread function which has the value one inside a circle with a radius of 7 pixels and 0 outside this circle. The values of point-spread function elements close to the radius are determined by the fraction of the pixel coverage of the circle. After blurring, noise is added such that the SNR became 30 dB. In [BL90], SNR is taken as the ratio of the variance of the blurred image (before noise addition) and the variance of the noise. The deblurring shown is quite successful. For the best parameter setting, the deblurred image might become identical to the original binary image after thresholding at the 50% level, except for some errors at the black/white boundaries [La90]. The deblurring in [BL90] is an iterative process. Using a conjugate gradient method, in the order of 30 iterations are required for a good result [La90]. Each iteration requires among other things two convolutions with the point-spread function of the blurring process, which has a size of 15*15 for the circle with radius 7, yielding in the order of 13500 multiply/addition operations per output pixel in total.

In order to qualitatively compare the performance of the combination of DFE and a receiver filter with the iterative approach of [BL90], we use a synthetic image given in Fig. 2. We blurred this image with a circular filter with radius 7, and added noise to obtain a SNR of 30 dB, see Fig. 3. We tried several values for $\sigma_a^2 \,/\, \sigma_v^2$ to optimise the receiver and feedback filter. A value of 25000 (44 dB) gave good equalisation results. We have taken here a receiver filter of size 25*25. The feedback filter size is taken as large as required in order to suppress all trailing ISI (39*20, because the blurring point-spread function is 15*15 in size and the feedback filter has to be causal). The use of larger filters resulted in a very limited improvement of the performance only. The result of the equalisation of the image in Fig. 3 is shown in Fig. 4. It contains 333 errors (0.5%). Decreasing the noise level of the image in Fig. 3 with 3 dB and processing this using DFE and the receiver filter resulted in an image identical to the original binary image.

On the basis of this limited experiment, we observe that for this application, the two methods show a similar performance. However, the number of operations required for the iterative method in [BL90] is much larger. As already indicated, some 13500 operations per pixel are required. Our method requires one convolution with a 25*25 point-spread function (receiver filter) and one with a size of 39*20 (feedback filter), yielding in the order of 1000 operations per output pixel in total.

Note that in our experiment, as well as in the experiments shown in Fig. 24 and 25 of [BL90], the original image was already *discrete*. In practise, the unblurred image (scene) is continuous. This is not according to the models used to develop both methods. The results for our method are then not as spectacular as shown in this section, but they are still quite useful as we will see in the next section. We expect that the results using the iterative method will also be less good if an image coming from a camera is deblurred.

3.2 Some 2-d experiments using camera input

In this section, we present some experiments using images obtained with a camera from text on paper. We also use an image of a clear sheet of paper, to correct the white level of images with text. After correction, we found that the intensities on the white parts of the paper still had a variation of about 10% peak-to-peak, due to the paper structure and system noise. The standard deviation of the intensities of the white parts is about 2.5% of the average intensity of these white parts. The white intensity is 200 and the black intensity is 25. This means that we have quite a high noise level ($\sigma_a^2/\sigma_v^2 \approx 300$, assuming equal probabilities for black and white). The gamma factor of the camera was set to 1.

We used two input images for the experiments using camera input. They are shown in Fig. 5 and 6. They are both white-corrected images of text. The lower-case characters in Fig. 5 are about 10-12 pixels high (x-height in typographical terminology). The characters in Fig. 6 are smaller. The lower case letters in this image are about 8-9 pixels high.

A *statically thresholded* image of input image with *large* characters is shown in Fig. 7. The threshold taken is the average of the saturated black and white levels. Note that this threshold is sometimes too high, because at the characters 'a' and '=' some of the white is thresholded to black. Note also that the threshold is sometimes too low, because the characters are somewhat thin. Both effects are due to the low-pass filter characteristic of the "channel" (camera + optics).

A *dynamically thresholded* [Be86] image of input image with *large* characters is shown in Fig. 8. A window size of 3*3 is used. The width of the characters is much more correct now. A problem with the dynamically thresholded images is that in low-contrast areas, the noise is thresholded, so that the paper and the inside of the black bar become noisy instead of constant white and black respectively.

We used the black bar in Figs. 5 and 6 to estimate the point-spread function of the imaging system. We assumed that the imaging system has a *Gaussian point-spread function*, with a possibly different sigma for horizontal and vertical. The step-response of a system with a Gaussian impulse response is the error function. Measuring a horizontal profile through a vertical edge of the bar and fitting an error function provided us with an estimation of the horizontal sigma. Likewise, we estimated the sigma for the vertical direction. The imaging system used a CCD camera (HTH MX). The images were taken in as interlaced images. The horizontal sampling clock was set such that square pixels were obtained. For this imaging system, the sigma found in horizontal direction is 1.26 (measured in pixels) and the sigma found for the vertical direction is 1.02. These values have been used for the "channel-model" in all our experiments. The error from this approximation turned out to be low; the maximum error relative to the largest point-spread function element was smaller than 2% for almost all point-spread function elements, with an exception of 4% for one element for both directions.

We equalised the camera plus optics assuming a Gaussian point-spread function with the measured sigmas, using a 9*9 receiver filter and a 6*11 feedback filter. The filters were optimised using 50 for σ_a^2/σ_v^2 at the receiver input. We used *binary DFE* to obtain the image in Fig. 9 from the image with *small* characters in Fig. 6 (Section 2.1). We took 200 for the white level and 25 for the black level. The values used in the system were first converted to get a zero D.C. level (refer to Section 2.3). That meant that 112.5 was subtracted from all the pixels in the input image before it was fed to the receiver filter. This offset was added to the final output to correct it.

Fig. 10 shows the image which is obtained using a *multi-level* quantiser using the image with *small* characters as input (Fig. 6). In this case, the quantiser Q produces 20 levels from 25 to 200. When this image is thresholded at D.C., we get the image in Fig. 11. If we compare Figs. 9 and 11, we see that the edges of the characters in Fig. 11 are much less ragged than those in Fig. 9. This is because each binary decision at character borders in Fig. 9 is in effect wrong, because these borders may cut right through the pixels. However, for Fig. 9, this does not lead to severe error propagation in the feedback loop. In Fig. 11, the decisions are more precise which reduces the error considerably. Using a higher σ_a^2/σ_v^2 (>500) for the optimisation of the filters resulted an increased error rate for both a binary and a multi-level quantiser and even in severe error propagation for binary quantiser. Much less error propagation was observed for a multi-level quantiser.

Note that the quality of the result using DFE for the image with *small* characters is similar to the quality of the result of static thresholding for the image with *large* characters.

Note that the output as presented in Fig. 10 is useful as well. The image in Fig. 10 is much sharper than the one in Fig. 6. Apparently, multi-level DFE can serve to *enhance images*.

4 Conclusions

In this paper, we have discussed the use of decision feedback equalisation (DFE) in order to deblur images from binary scenes.

DFE assumes that the original image is discrete and binary. If this assumption is met, we can obtain results which are similar to results obtained by iterative constrained restoration [BL90]. The computational cost, however, is much lower.

DFE, using a quantiser of *only two* levels is *not suited* for the processing of blurred images taken in with a camera from binary scenes. The resulting images show much noise near edges and sometimes much error propagation. This is because the assumption of a *discrete* original image (= binary scene) is not met.

DFE using a quantiser with more levels (more than 20 or so) in the feedback path, but with a thresholded output, yields *better quality binary images* than straightforward thresholding. In particular, lower case letters on paper, obtained with a camera and sampled such that they are 8-9 pixels high (x-height in typographical terminology), appear quite good in the binary output (Fig. 11). Straightforward thresholding of these characters when they are sampled such that they are even 50% larger (10-12 pixels) gives worse results (Fig. 7). Dynamic thresholding [Be86] with a window size of 3*3 gives good results for the larger characters as well, but has problems in low contrast areas.

A consequence of the previous conclusion is that using the grey-level output of a camera and DFE processing, an area that is *2 to 3 times larger can be imaged* with the same quality of the resulting binary image than when straightforward thresholding of the camera output is used. The required processing to do this comprises two spatial finite impulse response linear filters, a quantiser and an adder. To be able to use DFE, one has to know the point-spread function of the imaging system and the noise level. Furthermore, it is required to know the black and the white levels, which requires that all pixels are corrected for the sensitivity and lighting variations. This constraint can be relaxed somewhat by the use of a multi-level quantiser in the feedback path.

DFE using a quantiser with more levels (more than 20 or so) in the feedback path, without a final decision threshold can be used for *image enhancement*.

5 Acknowledgments

The author gratefully acknowledges the discussions with Jan Bergmans on equalisation methods and the careful reviewing of this paper by Jan Gerbrands and Bart de Greef.

6 References

[Be86] J. Bernsen, "Dynamic Thresholding of Grey-Level Images", *8th Int. Conf. on Patt. Rec.*, 1986, pp. 1251-1255.

[BL90] J. Biemond, R.L. Lagendijk, R.M. Mersereau, "Iterative Methods for Image Deblurring", *Proc. of the IEEE*, Vol. 78, no. 5, May 1990, pp. 856-883.

[BP79] C.A. Belfiore and J.H. Park, "Decision Feedback Equalisation", *Proc. of the IEEE*, Vol. 67, no. 8, Aug. 1979, pp. 1143-1156.

[La90] R.L. Lagendijk, personal communication.

[MS81] M.S. Mueller and J. Salz, "A Unified Theory of Data-aided Equalisation", *The Bell Systems Technical Journal*, Vol. 60, No. 9, November 1981, pp. 2023-2038.

[Qu85] S.U.H. Qureshi, "Adaptive Equalisation", *Proc. of the IEEE*, Vol. 73, no. 9, Sept. 1985, pp. 1349-1387.

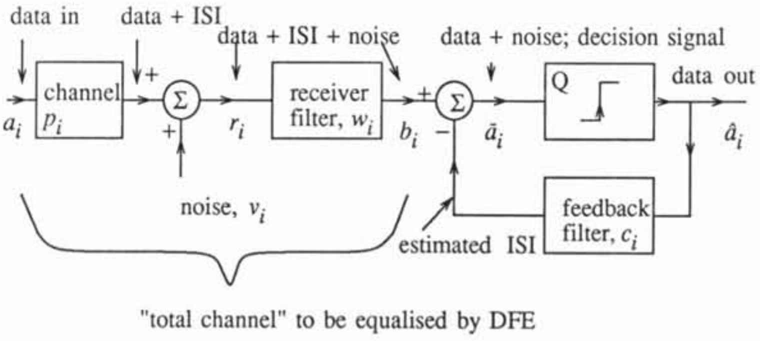
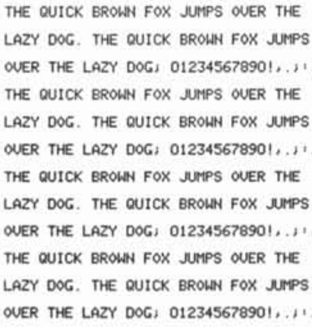


Fig. 1. Decision feedback equalisation using a receiver filter.



```
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
```

Fig. 2. Original binary image (256*256).

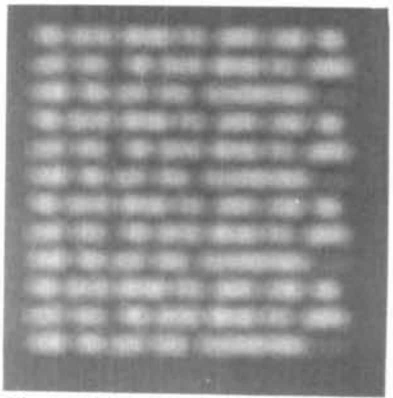
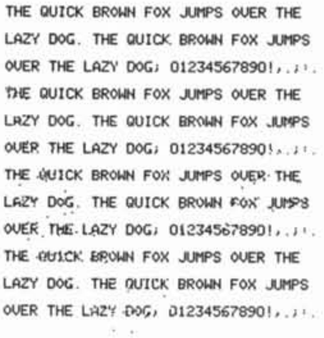
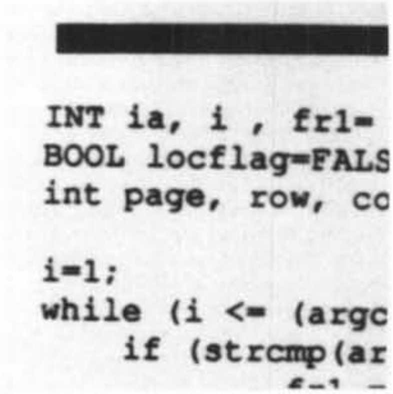


Fig. 3. Fig. 2 blurred with a circle of radius 7 and additive noise (SNR 30 dB).



```
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
THE QUICK BROWN FOX JUMPS OUER THE
LAZY DOG. THE QUICK BROWN FOX JUMPS
OUER THE LAZY DOG; 01234567890!,.;'.
```

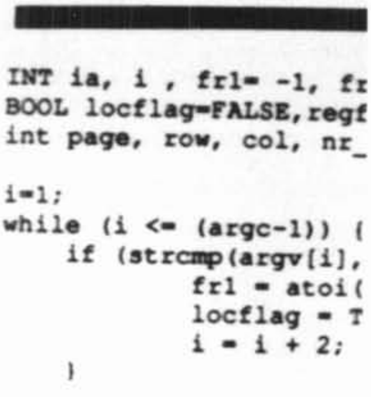
Fig. 4. Result of DFE and receiver filter processing on the image in Fig. 3.



```
INT ia, i , frl=
BOOL locflag=FALS
int page, row, co

i=1;
while (i <= (argc
    if (strcmp(ar
```

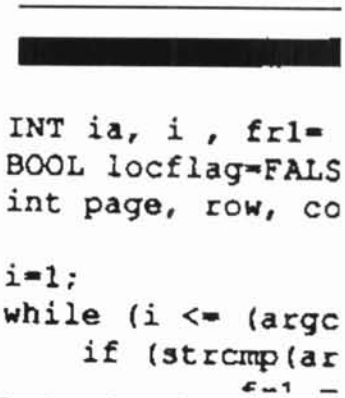
Fig. 5. Image with large characters coming from a camera (256*256).



```
INT ia, i , frl= -1, fr
BOOL locflag=FALSE,regf
int page, row, col, nr_

i=1;
while (i <= (argc-1)) {
    if (strcmp(argv[i],
              frl = atoi(
              locflag = T
              i = i + 2;
    }
```

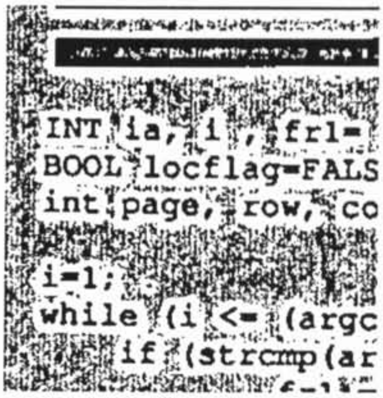
Fig. 6. As in Fig. 5, but with smaller characters (256*256 pixels).



```
INT ia, i , frl=
BOOL locflag=FALS
int page, row, co

i=1;
while (i <= (argc
    if (strcmp(ar
```

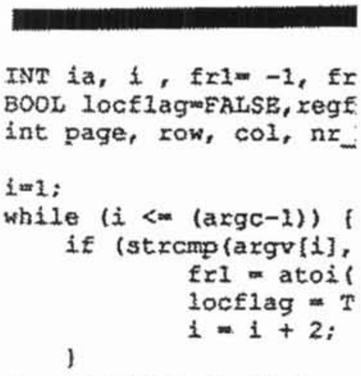
Fig. 7. Large character image (Fig. 5), thresholded with a fixed threshold at 113.



```
INT ia, i , frl=
BOOL locflag=FALS
int page, row, co

i=1;
while (i <= (argc
    if (strcmp(ar
```

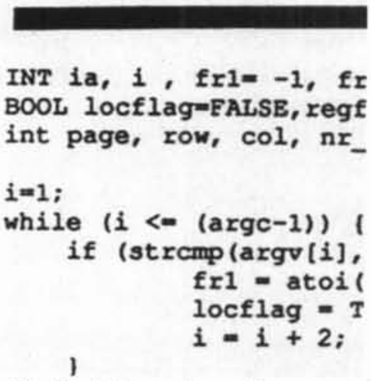
Fig. 8. Large character image (Fig. 5), dynamically thresholded using 3*3 window.



```
INT ia, i , frl= -1, fr
BOOL locflag=FALSE,regf
int page, row, col, nr_

i=1;
while (i <= (argc-1)) {
    if (strcmp(argv[i],
              frl = atoi(
              locflag = T
              i = i + 2;
    }
```

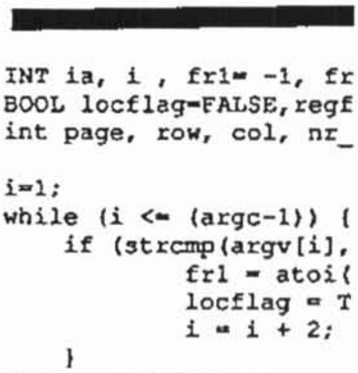
Fig. 9. Small character image (Fig. 6), processed using binary DFE ($\sigma_a^2/\sigma_v^2 = 50$).



```
INT ia, i , frl= -1, fr
BOOL locflag=FALSE,regf
int page, row, col, nr_

i=1;
while (i <= (argc-1)) {
    if (strcmp(argv[i],
              frl = atoi(
              locflag = T
              i = i + 2;
    }
```

Fig. 10. Small character image (6), processed using 20-level DFE ($\sigma_a^2/\sigma_v^2 = 50$).



```
INT ia, i , frl= -1, fr
BOOL locflag=FALSE,regf
int page, row, col, nr_

i=1;
while (i <= (argc-1)) {
    if (strcmp(argv[i],
              frl = atoi(
              locflag = T
              i = i + 2;
    }
```

Fig. 11. Image in Fig. 10, thresholded at D.C. ($\sigma_a^2/\sigma_v^2 = 50$).