Normalised Convolution Techniques

M. P. Foster

February 14, 2006

Abstract

This report introduces normalised convolution and some derived techniques for in-filling irregularly or sparsely sampled data. One of these techniques, adaptive normalised convolution, makes use of information on structure of the data in order to improve performance by adjusting filter parameters. This requires a gradient estimate, and for this reason, two methods of estimating the gradient of a irregularly or sparsely sampled data are descirbed. Result are evaluated in comparison with various traditional interpolation methods and the techniques presented in the report. The new techniques show good performance where traditional interpolation is unsatisfactory, demonstrating their usefulness on particularly sparse input data. Finally, some ideas for future research are presented.

Contents

Normalised Convolution Techniques

1.1 Introduction

Normalised convolution techniques are algorithms which operate on sparsely or irregularly sampled data, (or data sets with gaps) with a view to filling in the missing information. They are therefore a type of spatial interpolation.

By making use of available data and maps of data certainty, they are able to reconstruct data in areas where no samples are available with a good degree of accuracy, and with performance superior to traditional (gridding based) in-filling techniques, such as bilinear and bicubic interpolation.

Following this investigation, it should be possible to apply these techniques to sparse ionospheric TEC (total electron content) data in order to help reconstruct information over areas where no data are available.

This report will discuss four main normalised convolution techniques:

- **Standard normalised convolution (NC):**
  This is the simplest normalised convolution technique, and serves as a good introduction to the other more complicated techniques. See section 1.2.
- **Adaptive normalised convolution (ANC):**
  This is the most complicated of the techniques presented in this report. It makes use of information on the underlying image structure to form an improved reconstruction of the original data. See section 1.3. It also requires an estimate of the gradient of the image being operated on, which is the reason for the next technique:
- **Normalised differential convolution (NDC)** and **Differential of normalised convolution (DoNC):**
  These techniques construct sets of gradients using sparsely sampled data. See section 1.4.

Section 1.6 compares some interpolation schemes with standard and adaptive normalise convolution systems using figures of merit, and images.

Finally, section 1.7 discusses possible further work and conclusions.

1.2 Normalised Convolution

1.2.1 Introduction

Normalised convolution was first suggested by Knutsson and Westin in [KW93]. They introduced the idea that data *certainty* should be an important part of any operations where data points may be missing. They stress that there is an important difference between a measured value of zero, and a lack of data, and that these two situations should be separately represented. I.e. Data should consist of the actual data, and a map of the certainty of the data samples (which will generally be a binary matrix, the idea being that at each position, there is either a sample, or there isn't).

They also introduce a theory of basis functions, which states that an image can be expressed as a set of impulses, one for each pixel, where the strength of each impulse is the same as the value at each pixel.

When data are missing, the corresponding basis functions are also missing, and are therefore zero. They then go on to formulate the procedure for normalised convolution in tensor notation.

Since [KW93], various papers have been published which use normalised convolution for medical imaging applications, but only recently has normalised convolution research come back to the fore.

Farnback ([Far02]), for example, reformulates normalised convolution using a new type of transform[1]. This work then goes on to use normalised convolution as a basis for motion compensation and orientation estimation.

More recently, normalised has convolution been discussed in image processing terminology as opposed to tensor operations on data. In "Dealing with Irregular Samples" ([PP04]), Piroddi and Petrou discuss normalised convolution in such terms, and gives simple numerical examples illustrating the concepts.

1.2.2 Algorithm

The algorithm behind normalised convolution is very simple, involving just two convolutions and an element-wise division. The first convolution is defined by:

$$D(x,y) = f(x,y) * g(x,y) \tag{1.1}$$

Where $f(x,y)$ is the (sampled) input data. $g(x,y)$ is known as the *applicability function*, and which defines the localisation of the convolution by constraining the area over which it works.

In terms of basis functions, equation 1.1 spreads the basis functions around the image, according to $g(x,y)$.

The second convolution is defined by:

$$N(x,y) = c(x,y) * g(x,y) \tag{1.2}$$

Where $c(x,y)$ is the *certainty map* associated with the data $f(x,y)$. Equation 1.2 outputs a set of certainties associated with the first convolution. This could also be thought of as the certainty associated with the newly generated basis functions.

In order to normalise the first convolution, it is simply divided by the second:

$$\tilde{f}(x,y) = \frac{D(x,y)}{N(x,y)} \tag{1.3}$$

The output is therefore the first convolution, weighted by the confidence of the results generated, and is essentially a combination of linear and nearest neighbour interpolation of the original data.

1.2.3 Sample Output

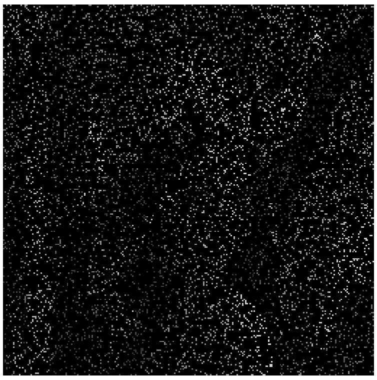
Figure 1.1(d) shows standard normalised convolution on the irregularly sampled test image Lenna.

1.2.4 Potential Problems

If the image to be reconstructed has gaps which are larger than the size of the filter being used in the NC, the output will have gaps. These gaps reduce the quality of the output image, and can produce nasty edge artifacts.

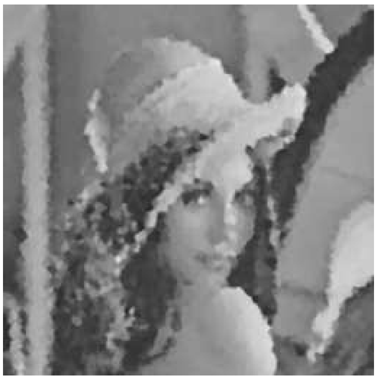
[1] Called a polynomial expansion

(a) Lenna



(b) Lenna: 90% of samples removed

(c) Lenna: After first convolution (sampled image with filter)



(d) Lenna: Reconstructed using NC, after having 90% of samples removed. The NC filter was a Gaussian of size 11.

Figure 1.1: Images of Lenna, before, sampled and after NC

To mitigate the problems that this causes, it is necessary to adapt the filter size at each point, in relation to the distance to the nearest sample. This leads to the concept of adaptive normalised convolution, which is discussed in section 1.3.

1.3 An Introduction to Adaptive Normalised Convolution (ANC)

As mentioned previously, normalised convolution aims to increase output quality by adapting to the input data, by choosing the smallest possible filter which encompasses at least one data point. This will ensure that the output image has no gaps, and should increase the quality of the output.

Unfortunately, however, things are not quite as straight forward as one would hope, and using only size adaptation provides very little improvement to the output image SNR over simply choosing a large enough filter, and using that over the entire input image. This could be for a variety of reasons, but is likely due to innaccuracies in available distance transform functions. Depending on the input image, the PSNR increase when using shape adaptive NC can range from zero to approxmimately 2.

In order to give a higher level of improvment over standard NC, an adaptive NC system must make use of information on the structure of the acutal image as well as the spacing between input samples.

Adaptive NC was first suggested by Pham and van Vliet ([PvV03]), and works by combining information

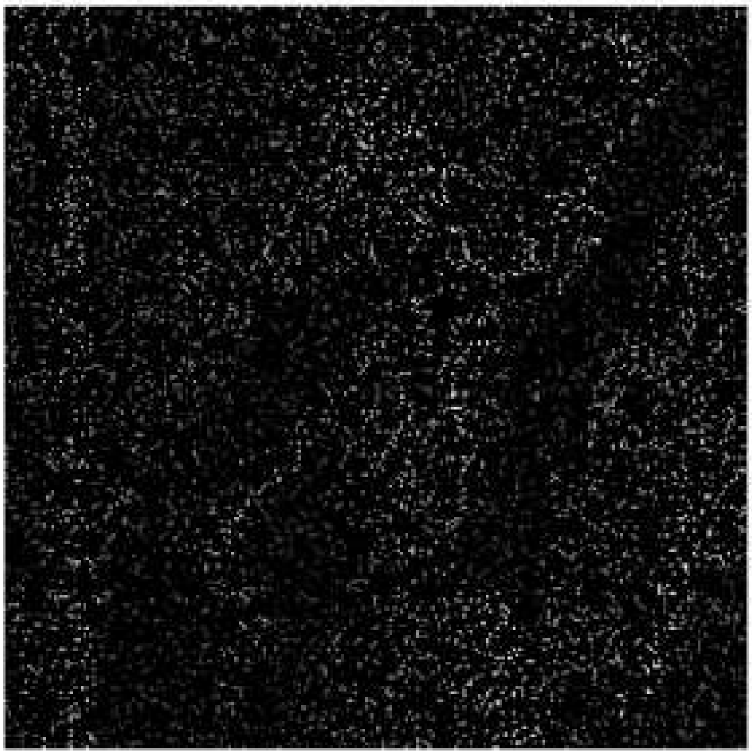


Figure 1.2: Lenna: Reconstructed using gaussian filter (size 3), after 90% data removal

on the distance between samples with information derived from local gradients in order to determine the best filter kernel for each point on the image. However, techniques for estimating the gradients of the input image must first be introduced. The following section introduces two gradient estimation techniques, and is followed by a return to the topic of ANC.

1.4 Gradient Calculation in Iregularly Sampled Images

1.4.1 Introduction

The two techniques introduced in this section provide a way of estimating the gradient of an image where some data are missing. They were first introduced by Knutsson and Westin, in their seminal paper [KW93], but are probably best summed up in [PP04], where various examples and comparisons with Sobel operators are given.

1.4.2 Derivative of Normalised Convolution (DoNC)

Applying differential operators to the normalised convolution (equation 1.2) is one (fairly simple) way of obtaining an estimate of the image gradient. It also happens to be fairly computationally simple.

Applying the differential operator to only the x axis gives:

$$\Delta_x \left(\frac{D(x,y)}{N(x,y)} \right) \equiv \frac{D_x(x,y) \times N(x,y) - N_x(x,y) \times D(x,y)}{N^2(x,y)} \tag{1.4}$$

Figure 1.3: Lenna: Reconstructed using shape adaptive NC.

Where:

$$D_x(x, y) = x.g(x, y) * f(x, y) \tag{1.5}$$

And:

$$N_x(x, y) = x.g(x, y) * c(x, y) \tag{1.6}$$

In the above equations, $x.g(x, y)$, is an edge enhancement filter which could be any arbitrary filter multiplied by a variable x. This efectively tilts the filter relative to the x axis. The example given in [PP04] is a raised cosine of the form (see figure 1.4):

$$g = cos^{\alpha}\left(\frac{\pi\sqrt{x^2 + y^2}}{8}\right) \tag{1.7}$$

This can then be extended to the y axis in a similar fashion, to give outputs consisting of the gradients in x and y, (Δ_x and Δ_y) in a similar fashion to the outputs of a Sobel edge detector.

1.4.3 Normalised Differential Convolution

Normalised differential convolution (NDC) is slightly more complex than DoNC, and works by constructing a set of filter matrices, (one for each point in the sampled image) which are then inverted, and used as a multiplier on the data. This gives good results, but is computationally more expensive than DoNC, because of the need to construct and invert a separate matric for each point.

The matrix, N_Δ is given by:

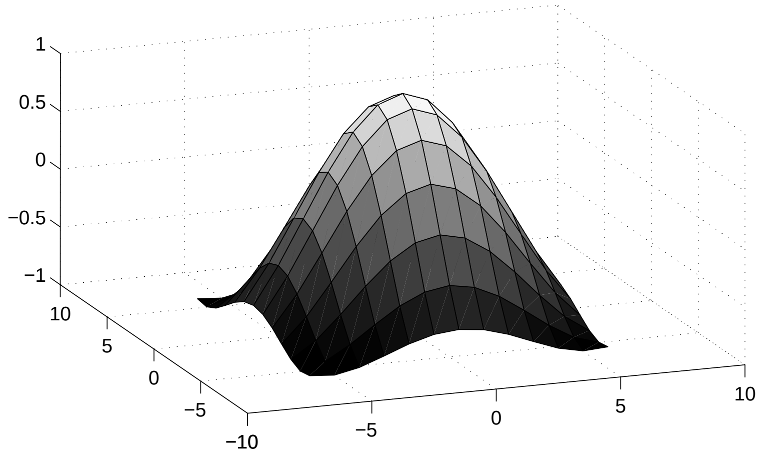


Figure 1.4: Raised cosine filter ($\alpha = 1$)

$$N_\Delta \equiv \begin{bmatrix} N_{xx} & N_{xy} \\ N_{yx} & N_{yy} \end{bmatrix} \tag{1.8}$$

The terms in equation 1.8 are defined by the following equations (dependence on x and y is implicit), which define the data certainty in x, y, and the diagonals (xy and yx). N_Δ is therefore based entireley on the data confidence map.

$$N_{xx} \equiv N \times ((x^2.g) * c) - N_x^2 \tag{1.9}$$

$$N_{yy} \equiv N \times ((y^2.g) * c) - N_y^2 \tag{1.10}$$

$$N_{xy} \equiv N_{yx} \equiv ((x.y.g) * c) - N_x \times N_y \tag{1.11}$$

The other term in the NDC is a vector, formed using the input data and differentiated in x and y.

$$D_\Delta = \begin{bmatrix} D_x \times N - N_x \times D \\ D_y \times N - N_y \times D \end{bmatrix} \tag{1.12}$$

The output for each pixel is then defined as:

$$\begin{bmatrix} \Delta_x \\ \Delta_y \end{bmatrix} = N_\Delta^{-1} D_\Delta \tag{1.13}$$

Like all of the other techniques in this report, the output is formed using some form of convolved data, divided by a confidence weighting.

As with DoNC, the filter g could take any form, provided that when multiplied by combinations of the variables x and y, they form directionally sensitive filters. Smaller filters give more localised edges, but the minimum filter size is dependent on distance between adjacent samples. In practice, the minimum filter size for a data set where 98% of samples have been randomly removed is about 8, with best results when the size approaches 11.

1.4.4 Sample Outputs

Figures 1.5 and 1.6 show DoNC and NDC edge magnitudes generated from an the Lenna image where 90% of data have been removed. Both functions use an identical 13×13 Gaussian mask. The output images are of a broadly similar quality, with the NDC having generally smoother output.

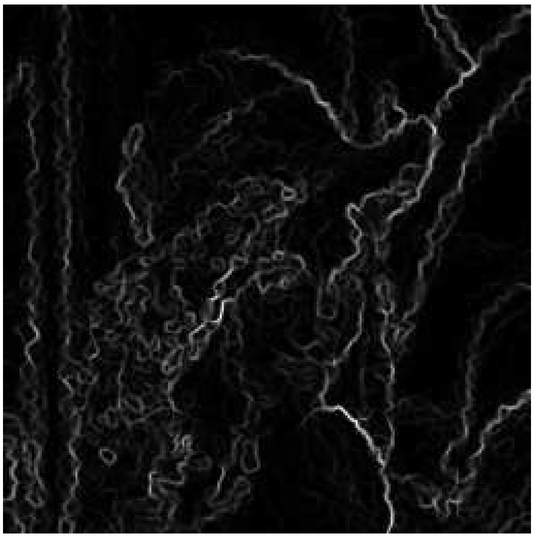


Figure 1.5: DoNC (90% data removal, Gaussian filter dimension 13)

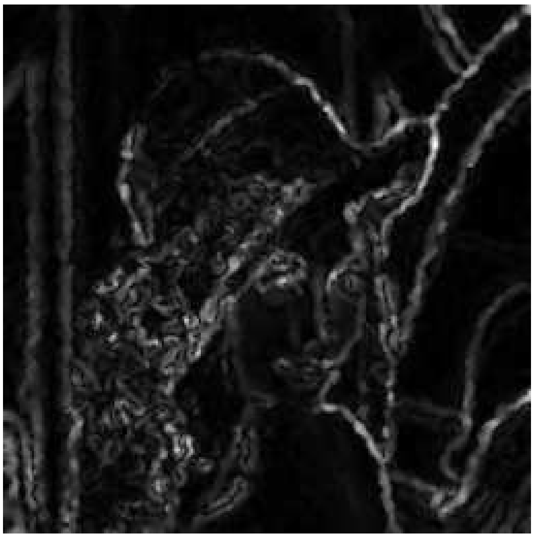


Figure 1.6: NDC (90% data removal, Gaussian filter dimension 13)

1.5 Adaptive Normalised Convolution Revisited

Adaptive NC works by making use of information on the underlying structure of the input image. This information, such as gradient and direction of local features allows a filter kernel to be sized and oriented to give the best possible output image SNR. Having explained how it is possible to estimate the gradient of an irregularly sampled image, the process of ANC can be examined in detail.

First, the gradient of the input image should be estimated. This can be determined using either NDC or DoNC, with DoNC offering comparable quality results at a lower computational cost.

Next, the gradients are multiplied together and smoothed using a Gaussian filter (where larger filter sizes are generally better) to give g_x^2, g_y^2 and g_{xy}.

All filters in ANC are usually (2D) Gaussian, which makes it easy to set the size in two dimensions by adjusting standard deviation values, and using the formula $d = 6\sigma + 1$.

After these new gradients have been computed, a *gradient scale tensor* (GST[2]) is produced for each pixel in the image. This is a two-by-two matrix, composed of pre-smoothed gradient products:

$$GST = \begin{pmatrix} g_x^2 & g_{xy} \\ g_{xy} & g_y^2 \end{pmatrix} \tag{1.14}$$

The GSTs eigenvalues are then computed, and from these two values the following metrics can be calculated: (λ_1 and λ_2 are the largest and smallest eigenvalues respectively.)

- The local anisotropy:

$$A = 1 - \frac{\lambda_1}{\lambda_2} \tag{1.15}$$

- The local energy:

$$E = \lambda_1 + \lambda_2$$

The eigenvalues and gradients also allow computation of the local gradient direction and orientation. The values lie in the range $\pm\frac{\pi}{2}$, and are given with respect to the x-axis.

- The local gradient direction is the direction associated with the largest eigenvalue:

$$\varphi_1 = tan^{-1}\left(\frac{\lambda_1 - g_x^2}{g_{xy}}\right) \tag{1.16}$$

- The local orientation is the direction associated with the smallest eigenvalue (this value is used to set the filter direction):

$$\varphi_2 = tan^{-1}\left(\frac{g_{xy}}{\lambda_1 - g_y^2}\right) \tag{1.17}$$

As discussed previously, the other variable used in determining the filter size at each pixel is the Euclidean distance from each pixel to its nearest neighbour. This is known as σ_a.

The filter standard deviations are then given by:

$$\sigma_u = C(1 - A)^{\alpha}\sigma_a \tag{1.18}$$
$$\sigma_v = C(1 + A)^{\alpha}\sigma_a \tag{1.19}$$

The terms C and α allow the degree of dependence on the local image structure and anisotropy to be adjusted. Values of $C = 1$, and $\alpha = 1.1$ give good results.

After the filter size and orientation for each pixel has been calculated, the standard normalised convolution procedure can be followed, where instead of using a fixed Gaussian filter, the filter for each point is

[2] See [vVV99] for more information on GSTs and their derivation.

constructed using the calculated data. This means that standard convolution routines must be modified in order to use different filters at each point.

The entire process can be seen in Figure 1.7, which shows how many different features of the filter may be customised, including optimal smoothing filters at each stage - in practice however, smoothing anything but the anisptropy and gradient products actually reduces the output quality.

Experimentation on changing filter sizes suggests that for the gradient products, a larger filter size is better, where as a more moderate filter size of 41 seems to be best for smoothing the anisotropy.

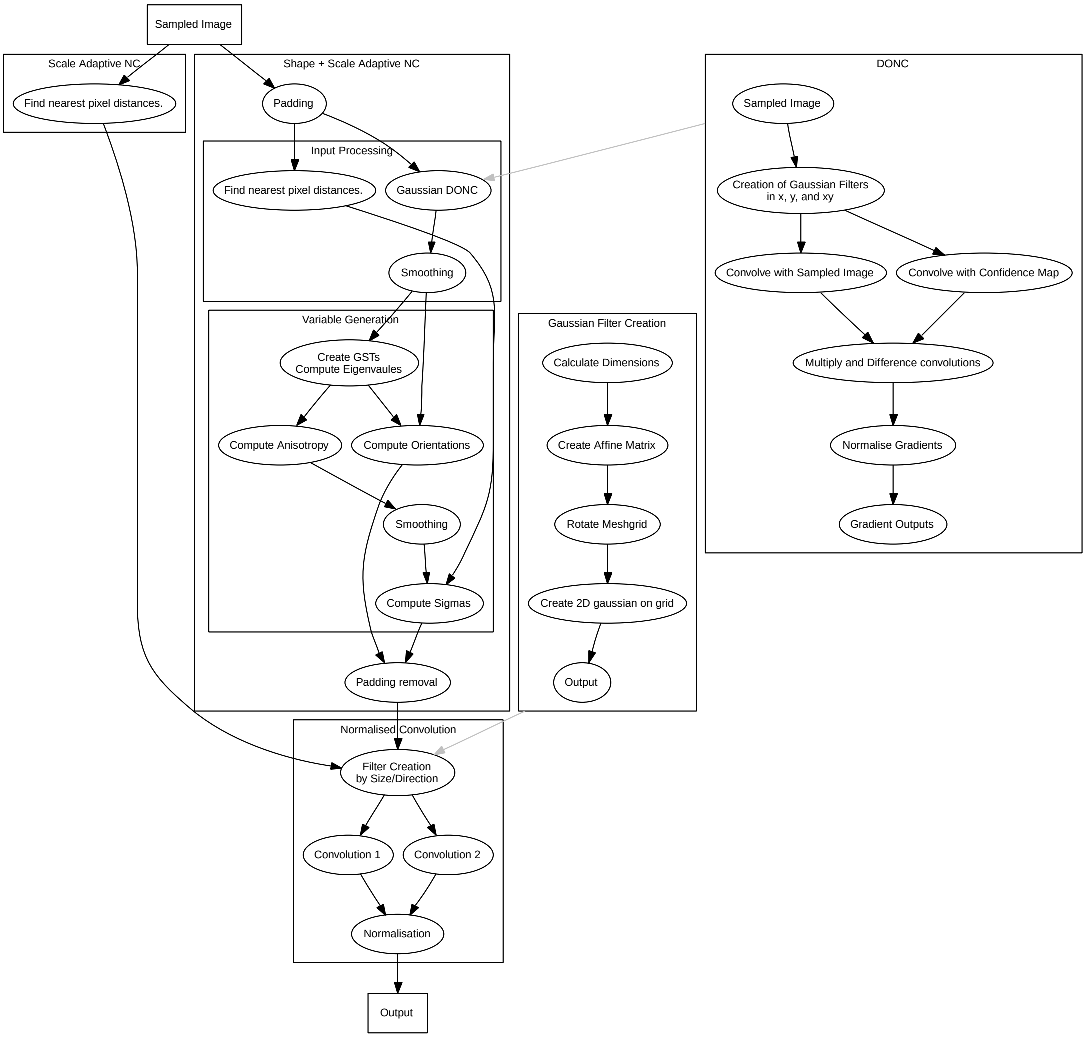


Figure 1.7: Flow chart showing NDC process.

1.5.1 Sample Outputs

Figures 1.1(b) and 1.8 show the test image Lenna, with 90% of samples randomly removed, and after reconstruction using ANC.

Figure 1.8: Lenna: Reconstructed using ANC (DoNC), from Figure 1.1(b). I.e 90% data removed.

RMS	Linear	Cubic	Nearest	NC	ANC (DoNC)	ANC (NDC)
90%	15.42	15.41	18.41	17.51	15.59	15.47
95%	19.59	19.69	22.20	22.68	18.81	18.73
98%	25.43	25.79	28.14	49.40	23.49	23.80
99%	31.92	32.33	32.05	79.79	27.57	27.20

Table 1.1: RMS values for 90-99% data removal.

PSNR	Linear	Cubic	Nearest	NC	ANC (DoNC)	ANC (NDC)
90%	48.74	48.75	45.66	46.53	48.55	48.68
95%	44.58	44.49	42.41	42.04	45.29	45.36
98%	40.05	39.80	38.29	28.51	41.43	41.20
99%	36.10	35.88	36.03	20.18	38.64	38.88

Table 1.2: PSNR values for 90-99% data removal.

1.6 Comparison of NC and ANC with Gridding Techniques

This section shows outputs generated by normalised convolution and adaptive normalised convolution alongside results generated using Matlab's *griddata* function, which interpolates data using various different methods.

The results clearly show that the ANC method start off with slightly lower quality output than the interpolation methods, but that the relative quality increases as more data are taken out. In figure 1.10(d) the difference in quality between ANC and the interpolation schemes is particularly apparent.

In figures 1.10(a)-1.10(d) the 'adaptive' output gradients were generated using DoNC, and are very similar to the NDC outputs (not shown).

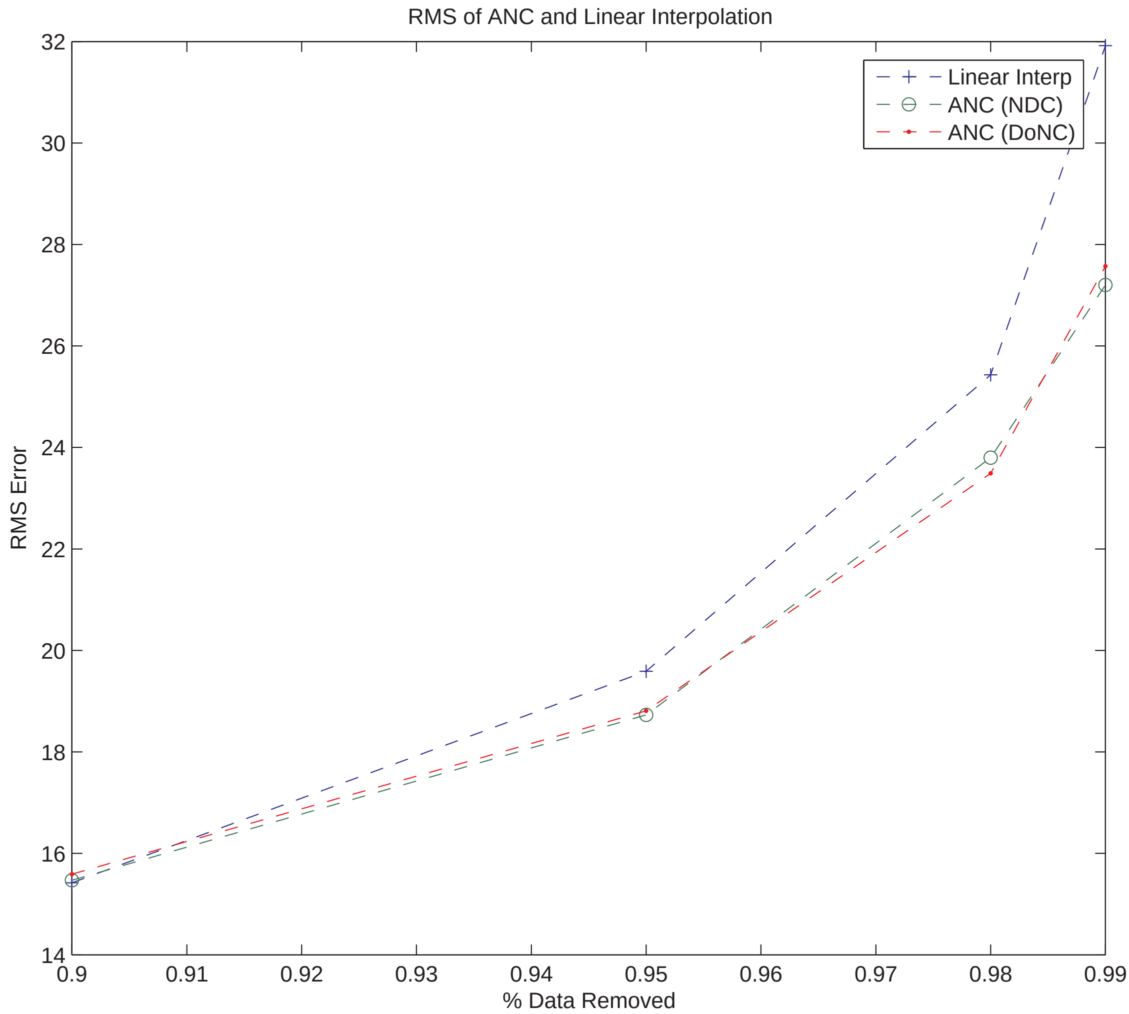


Figure 1.9: RMS values for ANC and Linear interpolation with varying data removal.

1.7 Conclusions and Future Research

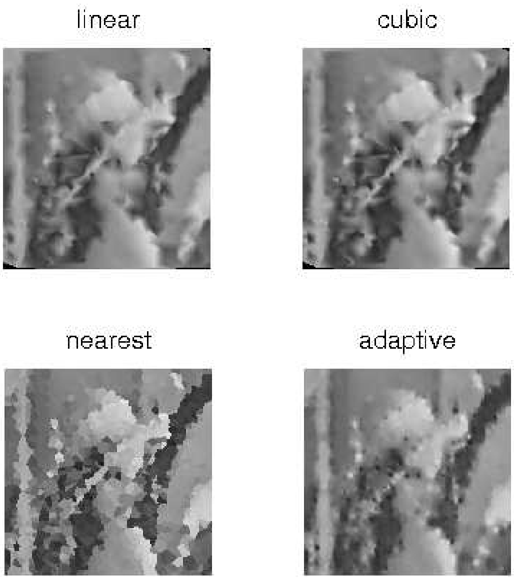
Normalised convolution, and techniques derived from it have been shown to be effective at filling in gaps in sparsely or irregularly sampled data, and giving results which are far better than traditional gridding techniques where data are particularly sparse.

The fact that normalised convolution techniques allow images to be reconstructed from very little data suggests that there may be applications in image compression. Further work could therefore be done in order to find out just how much information could be discarded, and from where, in order to reach a given compression rate, or PSNR.

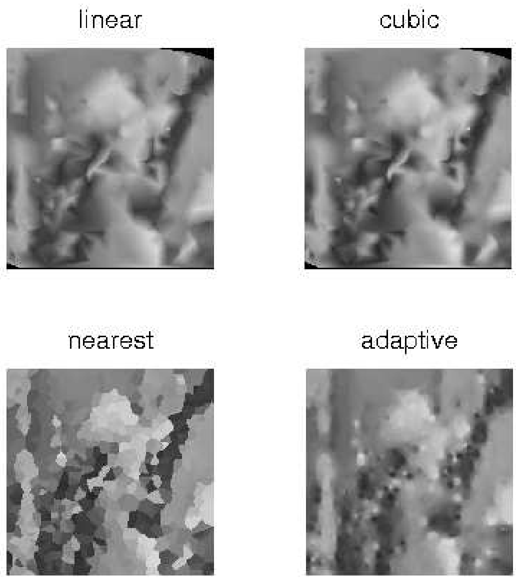
The main motivation for this investigation was in-filling TEC data. It should now be possible to attempt this, using the research described in this report. This will therefore become a major direction for future work.

(a) 90% data removal

(b) 95% data removal



(c) 98% data removal



(d) 99% data removal

Figure 1.10: Comparison of Matlab data gridding and ANC.

Bibliography

[Far02] G. Farneback. *Polynomial Expansion for Orientation and Motion Estimation*. PhD thesis, 2002.

[KW93] H. Knutsson and C.-F. Westin. Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data. In *Proceedings of Computer Vision and Pattern Recognition (Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition'93)*, pages 515–523, New York City, USA, June 16–19 1993.

[PP04] R. Piroddi and M. Petrou. *Dealing with Irregular Samples*, volume 132, pages 109–165. Elsevier Inc, 2004.

[PvV03] T. Q. Pham and L. J. van Vliet. Normalized averaging using adaptive applicability functions with applications in image reconstruction from sparsely and randomly sampled data. *Image Analysis, Proc.*, 2749:485–492, 2003.

[vVV99] L. J. van Vliet and P. W. Verbeek. Estimators for orientation and anisotropy in digitized images. In *Proc. 11th Scandinavian Conf. Image Analysis*, 1999.