Near optimal non-uniform interpolation for image super-resolution from multiple images

A. Gilman, D.G. Bailey

Institute of Information Sciences and Technology, Massey University, Palmerston North

Email: a.gilman@massey.ac.nz

Abstract

Non-uniform interpolation is a common procedure in image processing. A linear interpolation filter is generally a weighted combination of the inputs. Optimal filter coefficients (in a least squares sense) can be derived if the interpolated image is known beforehand. The weights of a general linear interpolation filter are independent of content and only depend on the relative positions of the available samples. The optimal coefficients are shown to be relatively independent of the content experimentally in absence and presence of noise, allowing non-uniform images to be interpolated using coefficients that have been optimised on a synthetic image. This results in a linear interpolation with computational complexity of the same order as nearest neighbour or bilinear, but with a near optimal performance.

Keywords: Image super-resolution, non-uniform interpolation, scattered interpolation.

1 Introduction

Non-uniform interpolation, also known as scattered interpolation, is a key step in image super-resolution from multiple images [1]. After the input low-resolution (LR) images are registered to the high-resolution (HR) grid, these can be combined (with appropriate offsets) to form a compound non-uniformly sampled image. An interpolation procedure can be applied to resample the compound image at the uniform positions of the high-resolution grid. There are a large number of interpolation methods exist, each making assumptions about the surface of the image function. The choice of a method depends strongly on the application specific requirements, as there is a trade-off between computational complexity, memory requirements, and optimality of the result. Sensitivity to the accuracy of registration procedure can also play a role in the selection.

The scope of this work was to look at global translational motion only, with low noise levels. Image degradation was assumed to be due to the camera point spread function only, constant in time and linear space-invariant for all input images.

Because of the assumption of global translational motion only, the non-uniform compound image is actually semi-uniform. Of course in this case the generalised sampling theorem [2,3] can be applied to reconstruct the exact high-resolution image, as long as average sampling rate is above the Nyquist rate. However, this procedure is computationally expensive and is sensitive to even low levels of noise. Our main interest is super-resolution methods with low computational complexity; therefore we consider this approach unsuitable.

The remainder of this article summarises a number of interpolation methods that can be implemented as digital filters, and develops a new, near optimal, method for computing the weights of a linear interpolation filter.

2 Near optimal interpolation

The simplest method for image interpolation is nearest neighbour interpolation [4]. For each point on the HR grid, the closest known LR pixel is selected and the value of that pixel is simply used as the value at the grid point. This method, therefore, implicitly assumes a piece-wise constant model for the image. It is the fastest of all interpolation methods as it considers only a single pixel – that closest to the grid point being interpolated.

Another simple and well-established method is bilinear interpolation. It can be applied to super-resolution in the following way [5]. To interpolate a given point on the HR grid, the closest LR pixel is used, along with its three neighbours from the same LR image, as pictured in figure 1. These three pixels are picked so that they are the next closest pixels to the point being interpolated. The four LR pixels are the vertices of a square, with the point being interpolated located inside the square. The value at the point is computed using a bilinear weighed sum of the four vertices. Bicubic interpolation can be implemented in a similar way, selecting 16 closest points and applying bicubic weights.

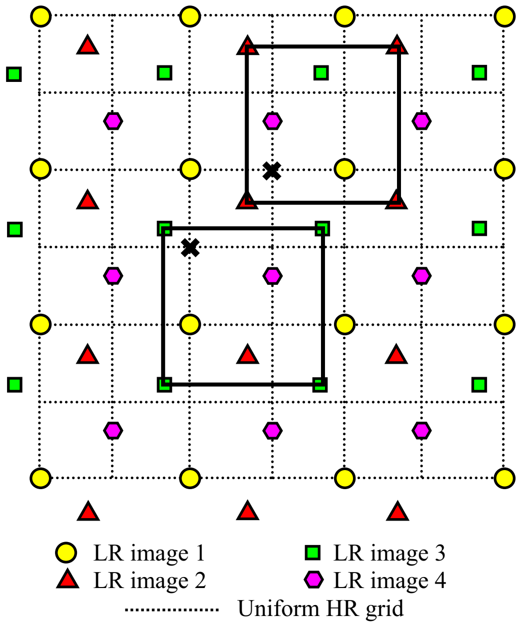


Figure 1: Increasing resolution by a factor of two, using bilinear interpolation. The four input samples used are generally not the closest samples available.

While this is fine for a single image, where there are many low-resolution images, three (or 15 in the case of bicubic) of the pixels that are used are not necessarily the closest, as these may be in other input images. Logically, the closest inputs to the output pixel are more likely to contain relevant information about the HR pixel value.

Instead of using four closest pixels from a single input image, the selection can be made from all pixels from all input images. A problem then arises of determining the weights for each of these inputs.

One approach is to determine the weights simply by a function of distance of each input sample from the output point (i.e. inverse distance, inverse distance squared, etc). This method has first been attributed to Shepard [6].

An alternative is to use Delaunay triangulation and then fit a plane or other type of surface to each triangle to interpolate an HR grid point inside that triangle [7].

Other methods also exist and can be generalised as follows. First, N input pixels around the point we want to interpolate are selected (or all input pixels falling inside a radius r around the output point). Then, the value for the output point is calculated as a weighted combination of these known pixels. The weights depend on the interpolation method. Since different methods would produce different weights, the obvious question is "which weights are the best?"

If the ideal HR image is known (i.e. ground truth), the weights that minimise the squared error could be computed. Such weights would be optimal in a least squares sense. The problem is that the desired output image is unknown; otherwise there would be no point to interpolate.

The weights of a general linear interpolation filter do not depend on the image content, but on the relative positions of the available samples. Therefore, it is hypothesised that the optimal weights should depend only weakly on the actual image content. If this is the case, then the optimal weights derived from one image should be close to optimal (thus near optimal) on other images with the same offsets. Hence, a synthetic image can be used to derive the weights, which are then applied to the input images.

In terms of implementation, such a method can be implemented as a two-dimensional finite impulse response filter, just as all the other methods described in this section. Hence, all these methods should be of similar computational complexity, apart from the overhead of calculating the coefficients for the "near optimal" method, which is run just once before the input images are processed.

3 Results and discussion

3.1 Experimental setup

To assess the performance of the above methods, it is necessary to have a ground truth high-resolution image. If the low-resolution images are simply captured, the ground truth is unknown. Hence we used a method similar to Bailey [8] to generate a number of LR images from a single very high-resolution image through a simple imaging model. Image 'beach' (as pictured in figure 2) was selected to be the test image – it has dimensions of 1700×1700 pixels. To form LR images, the source image was filtered using a 20×20 square box average filter to simulate area integration, shifted by random (integer pixel) offsets, then down-sampled by a factor of 20. Finally, Gaussian noise was added to simulate the effects of various noise sources within the process. To form the HR image, the source image was filtered using the same filter, but down-sampled only by a factor of 10. Note that the high-resolution image is blurred to the same degree as the low-resolution images. This enables the performance of interpolation methods alone to be investigated, hence deblurring is left out. In addition, the exact known offsets were used to ensure there is no misregistration of the low resolution images.

Figure 2: Image 'beach'.

The down-sampling procedure resulted in offsets that are integer multiples of 0.05 of a pixel. Four LR images were used to super-resolve a single HR image by a factor of two. Simulating every possible combination would be very time consuming; hence, a Monte Carlo simulation was used consisting of 10,000 runs with randomly selected offsets. On each run the four randomly-offset low resolution images were combined into a single super-resolved image using one of the interpolation methods. The output image was compared to the ground truth image and the mean square error was calculated.

3.2 Experimental results

The results can be interpreted using an inverse cumulative distribution function (iCDF), also known as percent point function or quantile function [9]. To form this function, the errors from all the runs are ranked in ascending order and plotted, with the probability of 0.0001 associated with the smallest error and probability of 1.0 associated with the largest error. The probability, therefore, gives the probability that the interpolation error is less than the associated error.

The inverse cumulative distribution functions of the errors for nearest neighbour, bilinear and optimal interpolations can be seen in figure 3. This plot shows the expected performance for a particular percentile. So for example for 50^{th} percentile (the median), nearest neighbour interpolation is expected to yield MSE below 16.4×10^{-5}.

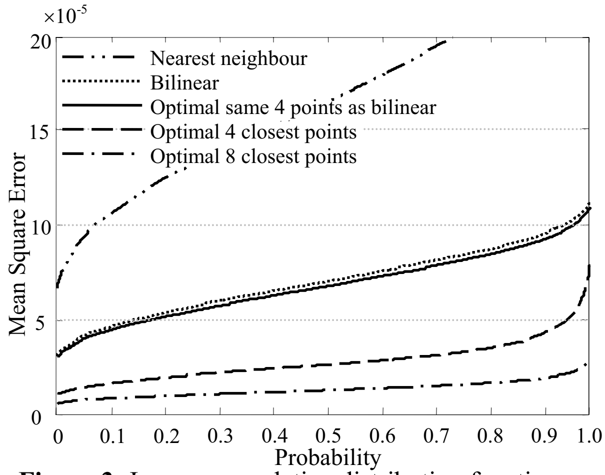
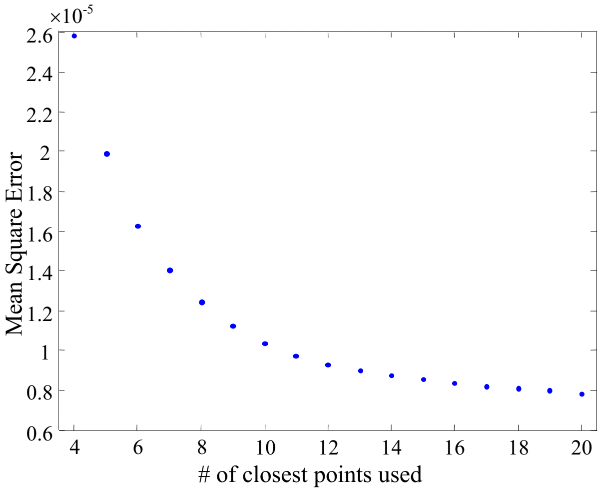


Figure 3: Inverse cumulative distribution functions (percent point functions) for nearest neighbour, bilinear, and optimal interpolations of image Beach.

It can be seen that the bilinear interpolation performs significantly better than the nearest neighbour interpolation. This result is intuitive, as bilinear interpolation uses more input information (four input points in comparison to one input point). The line just under bilinear corresponds to the optimal result that could be achieved if the same four points as used by the bilinear are utilised. For a given image, and a selection of input pixels used to perform the interpolation, this optimal method will give the smallest mean squared error (MSE) that can be obtained using a linear interpolation filter and can be used as a benchmark to compare other methods. The proximity of bilinear to optimal result is a good indication that bilinear interpolation is always stable and utilises the input information extremely well.

However, as already mentioned, the four input points used by the bilinear are not the ideal choice for super-resolution. Points further away from the desired location carry less relevant information, so ideally we want to use closest possible input points. This is confirmed by figure 3: optimal interpolation using four closest points yields a factor of two improvement over using the same points as the bilinear interpolation.



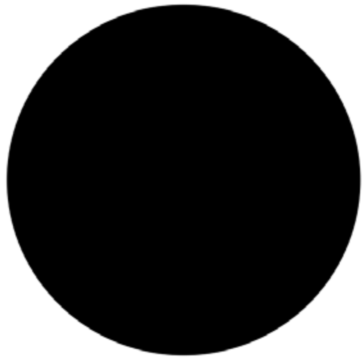


Figure 4: Images 'sleep' and 'disk'.

Figure 4 shows the optimal performance (median of iCDF) using between 4 and 20 closest points. It is clear that although the performance improves with an increasing number of points, the gains from using additional points decreases with points further away from the point being interpolated. Between eight and ten points seem to be a good compromise between accuracy and computational effort.

Two very different images were used to simulate the near optimal coefficients to test the hypothesis that coefficients derived from one image should work on another image. These are pictured in figure 5.

Figure 5: Images 'sleep' and 'disk'.

Image 'sleep' was chosen because it differs from image 'beach', but has similar statistics. Image 'disk' was chosen because it has significantly different statistics to have some idea how image content affects the results. It is also a synthetic image that can be used to calculate the coefficients for an arbitrary image.

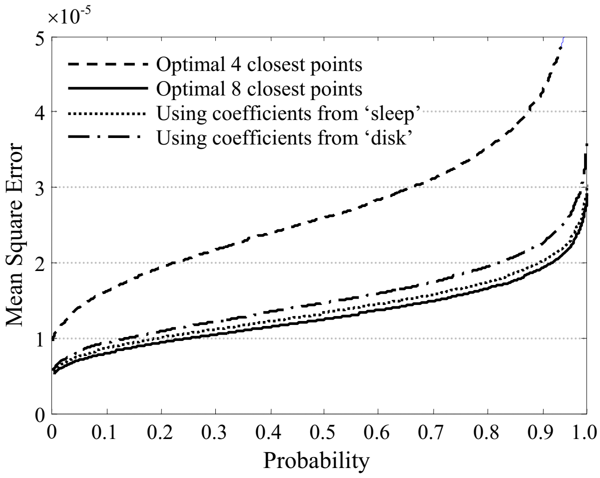


Figure 6: iCDFs of optimal interpolation and interpolation using coefficients optimised on images 'sleep' and disk' (using 8 closest points) and optimal using 4 closest points for comparison.

Eight closest points were chosen to be used to test the hypothesis. Figure 6 shows that using the coefficients optimised on images 'sleep' and 'disk' generate results very close to that of optimal. Image 'disk' has very different statistics, but it contains edges of all directions. This is possibly the reason why coefficients optimised on it offer reasonable interpolation results. These results are tabulated in Table 1.

Table 1: 1st, 2nd, and 3rd quartiles of iCDFs plotted in figure 6.

Method (8 points)	**1st quartile ($\times10^{-5}$)**	**2nd quartile ($\times10^{-5}$)**	**3rd quartile ($\times10^{-5}$)**
Optimal	0.99 **(100%)**	1.25 **(100%)**	1.56 **(100%)**
Coefficients from 'sleep'	1.06 (107%)	1.32 (106%)	1.64 (105%)
Coefficients from 'disk'	1.15 (116%)	1.45 (116%)	1.83 (117%)

The previous experiment was performed in the absence of noise. This is seldom true in practical imaging systems. Even if most sources of noise are minimised, there is still quantisation noise. For a typical 8-bit system this would have a standard deviation of 0.29 of a greyscale level. To check whether noise has any significant effect on the predicted coefficients, we super-resolved the 'beach' image at different levels of additive white Gaussian noise using coefficients optimised on the same image but without noise.

Table 2: Interpolating 'beach' at different noise levels using coefficients optimised on the same image at those noise levels and coefficients optimised without noise.

Median MSE ($\times10^{-5}$) / Noise s.d.	**Optimised at that noise level**	**Optimised with no noise**
0	1.25	1.25
0.5	1.76	1.90
1	2.67	3.58
2	4.90	10.4
3	7.35	21.9
4	9.91	37.8

Second column of Table 2 shows the performance of the optimal coefficients at various levels of noise. As expected, the MSE increases with the noise standard deviation. Third column shows the performance of coefficients optimised without noise. It can be seen that the relative performance deteriorates as the noise is increased.

Addition of Gaussian noise to the LR image formation model is investigated as a possible way of improving performance with noisy inputs. Five sets of coefficients are created, each optimised on image 'beach' using noise levels of zero, one, two, three, and four. Each set of coefficients is applied to inputs with various levels of noise (between zero and 4) and the results are plotted in figure 7.

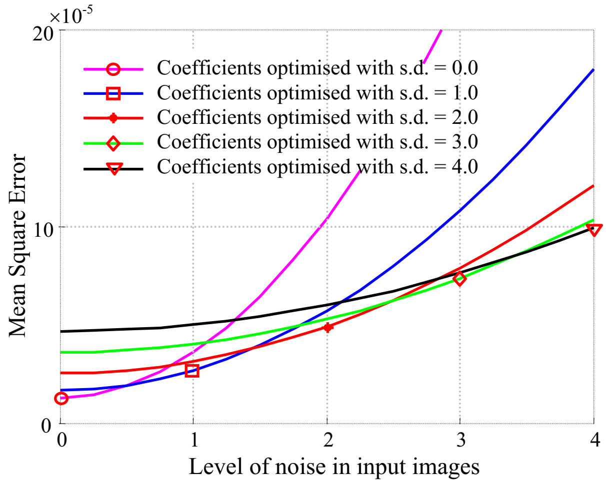


Figure 7: Performance of optimal interpolation on image 'beach' in the presence of noise using coefficients optimised at various levels of noise.

Figure 7 shows that for each input noise level, the best performance is achieved if the coefficients are optimised on the same level of noise. Now the same test can be applied to optimised coefficients on a different image. The procedure is exactly the same, only image 'sleep' is used to optimise the coefficients.

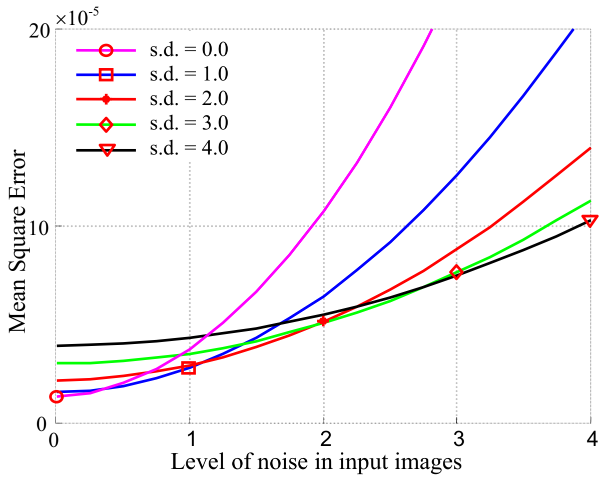


Figure 8: Performance of optimal interpolation on image 'beach' in the presence of noise using coefficients optimised on image 'sleep' at various levels of noise.

Figure 8 shows a similar trend with the coefficients being optimised on a different image. Hence, if the level of the noise in the input images can be estimated [10,11], the same amount (in the case of using image 'sleep' it can be seen that a slightly higher noise level is required) of noise can be used to optimise the coefficients. It can also be noticed that using coefficients estimated on noise levels within ±1 of the input noise level yield satisfactory results, so the input level of noise needs to be estimated only approximately.

4 Conclusions

Non-uniform interpolation is a common procedure in image processing. This paper has focused on a new method of deriving the weights of a linear interpolation filter, which are optimal in a least squares sense. Based on the observation that the weights of a general linear interpolation filter depend only on the relative positions of the available samples, it was hypothesised that the optimal weights derived on one image would be near optimal on other images.

Experimentation showed that the optimal weights, derived through minimising the squared error between a known high-resolution image and a set of synthetically-created low-resolution images, are relatively independent of image content. Hence, weights optimised on a known image, can be used to interpolate an unknown image with the samples positioned in the same place as the known image.

While, in general, the desired high resolution pixel values are not available to calculate the optimal weights, this opens the possibility for near optimal interpolation using a synthetic image to derive the coefficients.

It was shown that in the presence of noise, the coefficients optimised at the same noise level as the input are likely to yield better results.

A future direction of work is to find an analytical way of deriving near optimal coefficients, hence decreasing the overhead. More experiments are planned to be completed to check the method on other test images and to use a variety of synthetic images to generate the weights.

5 References

[1] S.C. Park, M.K. Park, and M.G. Kang, “Super-resolution image reconstruction: a technical overview,” *IEEE Signal Processing Magazine*, vol. 20, pp. 21-36, May 2003.

[2] A. Papoulis, “Generalized sampling expansion,” *IEEE Transactions on Circuits and Systems*, vol. 24, pp. 652-654, 1977.

[3] J. Yen, “On nonuniform sampling of bandwidth-limited signals,” *IRE Transactions on Circuit Theory,* vol. 3, pp. 251-257, 1956.

[4] N. Nguyen, P. Milanfar, and G. Golub, “A computationally efficient superresolution image reconstruction algorithm,” *IEEE Transactions on Image Processing*, vol. 10, pp. 573-583, Apr 2001.

[5] M. Kunter, K. Jangheon, and T. Sikora, "Super-resolution mosaicing using embedded hybrid recursive flow-based segmentation," in *The Fifth International Conference on Information, Communications and Signal Processing*, 2005, pp. 1297-1301.

[6] D. Shepard, “A two-dimensional interpolation function for irregularly-spaced data,” in *Proceedings of the 1968 23rd ACM national conference*, 1968, pp. 517-524.

[7] S. Lertrattanapanich, and N. K. Bose, “High resolution image formation from low resolution frames using Delaunay triangulation”, *IEEE Transactions on Image Processing*, vol. 11, pp. 1427-1441, Dec 2002.

[8] D.G. Bailey, A. Gilman, and R. Browne, "Bias characteristics of bilinear interpolation based registration," in *IEEE Region 10 Conference (IEEE Tencon, Melbourne, Australia, (21-24 November, 2005),* 2005.

[9] G.R. Shorack, *Probability for Statisticians*, New York: Springer, 2000.

[10] K. Rank, M. Lendl, and R. Unbehauen, “Estimation of image noise variance,” *IEE Proceedings – Vision, Image and Signal Processing*, vol. 146, pp. 80-84, 1999.

[11] G.E. Healey, and R. Kondepud, “Radiometric CCD camera calibration and noise estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 16, pp. 267-276, 1994.