

Super Resolution for Automated Target Recognition

Maksim Levental

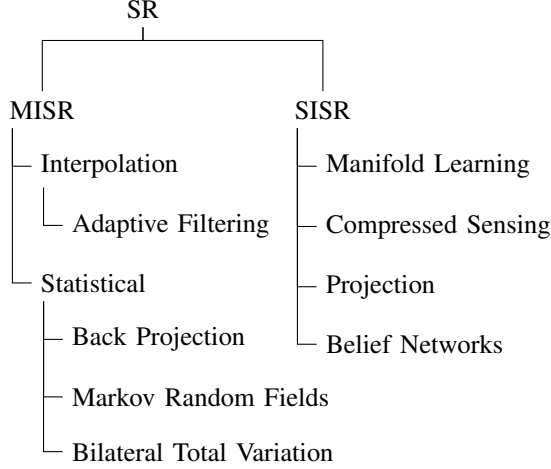


Fig. 1: Taxonomy of classical SR techniques

Abstract—Super resolution is the process of producing high-resolution images from low-resolution images while preserving ground truth about the subject matter of the images and potentially inferring more such truth. Algorithms that successfully carry out such a process are broadly useful in all circumstances where high-resolution imagery is either difficult or impossible to obtain. In particular we look towards super resolving images collected using longwave infrared cameras since high resolution sensors for such cameras do not currently exist. We present an exposition of motivations and concepts of super resolution in general and current techniques, with a qualitative comparison of such techniques. Finally we suggest directions for future research.

1 INTRODUCTION

2 BACKGROUND

3 CLASSICAL ALGORITHMS

3.1 Registration

Figure 1 lays out a rough taxonomy of classical SR algorithms. We cover algorithms from each "genus" and others that don't neatly fit into the taxonomy.

3.2 Interpolation

Suppose that H_k is linear spatial¹ and time invariant. Suppose further that A_k is affine. Then $H := H_k$ commutes with A_k [1] and equation 1 becomes

$$\begin{aligned}
 X_k &= (D_k \circ A_k \circ H)(Y) + \varepsilon \\
 &= (D_k \circ A_k)(H(Y)) + \varepsilon \\
 &= (D_k \circ A_k)(V) + \varepsilon
 \end{aligned} \tag{1}$$

¹In analogy with Linear Time Invariant (i.e. linear and constant in space).

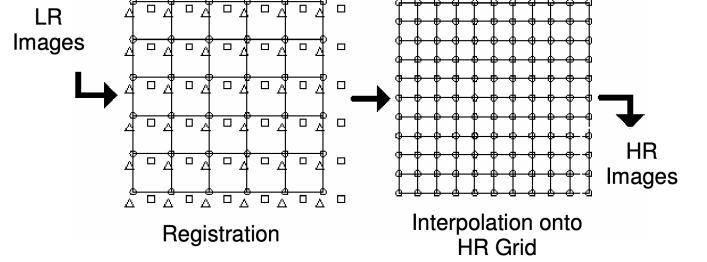


Fig. 2: LR image registration on an HR grid[2]

This naturally suggests interpolation in order to recover V (since X_k , in this framing, is simply shifted samples of V). Note in this context we use interpolation very broadly, i.e. to connote filling in missing values using neighboring (in some sense — not necessarily geometrically) values. This class of techniques proceed by first registering images on a high resolution grid (see figure 2) then interpolating at the "missing" pixels in the HR grid to recover V , and finally denoising and deconvolution (of H) to recover Y . Since in general consecutive X_k have non-uniform shifts (relative to X_0) the interpolation is non-uniform and improvisations on this theme use various weighting schemes for adjacent LR pixels².

For example Alam et. al[3] uses weighted nearest neighbors: for every pixel to be interpolated the three nearest pixels are weighted inversely by their distance (according to HR grid distance) and then their weighted sum is assigned to that pixel. This non-uniform interpolation is then followed by application of a Wiener filter whose design is informed by the OTF of the particular imaging system they study (which they do not estimate i.e. they assume they can model accurately). Lertratanapanich et. al[4] base their algorithm on interpolants which require knowledge of gradients (e.g. splines) and mediate the non-uniform sampling by using a weighted average (by area) of those gradients in adjacent Delaunay cells; to be precise they produce a Delaunay triangulation of all LR pixels and compute the gradients (see figure 3) according to

$$\begin{aligned}
 \vec{n} &= \sum_{j=1}^k \frac{A_j \vec{n}_j}{A} \text{ where } A = \sum_{i=1}^k A_i \\
 \frac{\partial z}{\partial x} &= -\frac{n_x}{n_z} \text{ and } \frac{\partial z}{\partial y} = -\frac{n_y}{n_z}
 \end{aligned}$$

Unfortunately this intricate solution is not robust to noise in real images.

A more sophisticated method for non-uniform interpolation uses parametric models for the auto-correlation between LR

²An LR pixel is one sampled from an LR image and embedded in an HR grid. An HR pixel is a pixel in an HR grid.

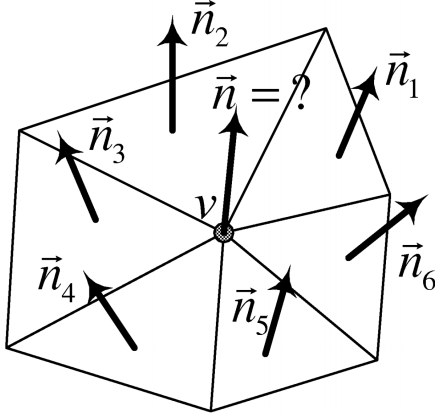


Fig. 3: Delaunay triangulation for fitting splines at LR pixels[4]. v is an LR pixel. Note that v is at z equal to the pixel value

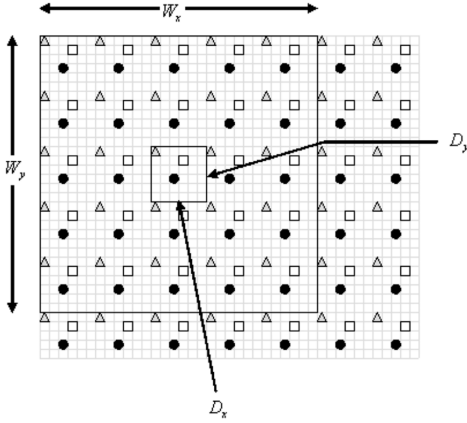


Fig. 4: Wiener filter super resolution estimation window of dimension $D_x \times D_y$ and observation window of dimension $W_x \times W_y$ [5]

pixels and the cross-correlation between LR pixels and interpolated pixels to estimate wiener filter weights[5]. These weights are then used to average nearby pixel values. The algorithm operates on a sliding *estimation window* whose dimensions D_x, D_y are chosen such that the effective sampling rate exceeds the Nyquist rate for a given ρ_c . The pixel values for the estimation window are a function of the wiener filter weights of nearby LR pixels within an *observation window* whose dimensions W_x, W_y are an integer multiple of D_x, D_y (see figure 4). The weights w are defined as the solution to the minimum mean squared error (MMSE) filter problem, i.e. the finite impulse response (FIR) wiener filter:

$$w = R^{-1}p \quad (2)$$

where R is the auto-correlation of the LR pixels in the observation window and p is the cross-correlation between the pixels to be estimated and the LR pixels. Then R and p are both constructed by sampling a parametric model that weights pixels in the observation window according to distance: R is constructed by sampling from

$$C_1(r) = \sigma_d^2 \rho^r * G(r) \quad (3)$$

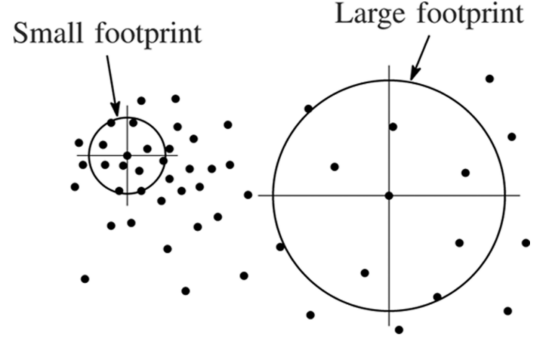


Fig. 5: Kernel footprint as a function of sample density[6]

and p is constructed by sampling from

$$C_2(r) = \sigma_d^2 \rho^r * G(r) * G(-r) \quad (4)$$

In the case of R , r is distance on the HR grid, σ_d is related to the empirical variance of all LR pixels in a given observation window and $G(r)$ is a smoothing kernel (e.g. gaussian). Thus by evaluating C_1 for all $r = r(n_1, n_2)$ distances between LR pixels n_1, n_2 we can construct R . Similarly for p , $r = r(m, n)$ is the distance between pixel-to-be-estimated m and LR pixel n . Note that R is an $N \times N$ matrix where $N = KW_xW_y/D_xD_y$, i.e. how many LR pixels there are in the observation window, and p is an $N \times 1$ column vector uniquely computed for each pixel in the estimation window. The scheme is effective but suffers from issues with the spatial isotropy of the auto-correlation and cross-correlation models.

One of the most sophisticated of these non-uniform interpolation schemes employs the kernel regression framework and *steering kernels*[6]. In this context we start with all X_k registered to a common HR grid and consider pixel values $Y_i = Y(x_i)$ at pixel coordinates $x_i = (x_{i1}, x_{i2})$ as the measured data pairs (x_i, Y_i) . Recall that kernel regression frames the estimation problem as

$$Y_i = Z(x_i) + \varepsilon \quad (5)$$

where Z is the to-be-estimated *regression function* that "predicts" Y as a function of x . Then the Nadaraya–Watson estimator (NWE)[7] \hat{Z} for Z is

$$\hat{Z}(x) = \frac{\sum_{i=1}^P K(x - x_i) Y_i}{\sum_{i=1}^P K(x - x_i)} \quad (6)$$

where P indexes over all pixels in the HR grid and K is a *kernel function* whose purpose is to decay the contribution of x_i if it's in some sense far from x . In conventional kernel regression K might be any non-negative, symmetric, unimodal[8] function with augmented with an additional h parameter that controls the "bandwidth" or "footprint" of the kernel, i.e.

$$K_h(x - x_i) = \frac{1}{h} K(h^{-1}(x - x_i)) \quad (7)$$

This bandwidth parameter h can be generalized to a *smoothing kernel* H in order to make $K = K_H$ adaptive to the local structure of the pixels, e.g. to have larger footprints

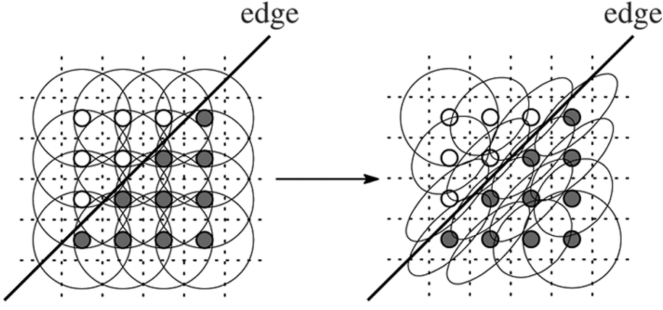


Fig. 6: Adapting kernel shape as a function of local directed structure[6]

in sparsely sampled regions and have smaller footprints in densely sampled regions (see figure 5). Ultimately though it is desirable to have kernels that can adapt to directed structure in the image, i.e. "steerable" kernels that filter strongly along an edge and weakly across an edge. This is accomplished by, for example, using a Gaussian as the kernel:

$$K_{H_i}(\mathbf{x} - \mathbf{x}_i) \propto \frac{\exp\{-(\mathbf{x} - \mathbf{x}_i)^T H_i^{-1}(\mathbf{x} - \mathbf{x}_i)\}}{\sqrt{\det H_i}} \quad (8)$$

and identifying H_i with $\nabla^2 Z(\mathbf{x}_i)$ (since gradients capture edge structure). An estimate \hat{H}_i of $\nabla^2 Z(\mathbf{x}_i)$ can be obtained by looking at covariances of empirical gradients (i.e. the HR grid registered image convolved with a difference filter). Unfortunately this is a naive estimate that is often rank deficient or unstable (both leading to instances where \hat{H}_i isn't invertible). One solution is to parameterize H_i :

$$H_i = \gamma_i U_{\theta_i} \Lambda_{\sigma_i} U_{\theta_i}^T$$

where U_{θ_i} is a rotation matrix, $\Lambda_{\sigma_i} = \text{diag}(\sigma_i, \sigma_i^{-1})$ is an "elongation" matrix, and γ_i is a scaling parameter, with each of $\gamma_i, \theta_i, \sigma_i$ estimated from the data in a more robust way. Another solution is to use a multi-scale Kalman filter to estimate H_i [9].

In general non-uniform interpolation techniques are intuitive and typically (relatively) computationally efficient but they assume an unrealistic observation model (namely that affine flow).

3.3 Estimation

Statistical estimation methods cast SR as an inference problem. One of the earliest successful SR algorithm[10] proposed an iterative algorithm inspired by the back-projection method commonly used to reconstruct 2-D objects from 1-D projections in computer-aided tomography. Recall that we assumed that motion and blur commute (see eqn. 1). Then the idea is to take the current estimate of the HR image \hat{V}^i , see if after motion and down-sampling it is near the LR samples X_k , and add a correction when it is not:

$$\hat{V}^{i+1} = \hat{V}^i + \sum_k (D_k \circ A_k)^{-1} \left((D_k \circ A_k)(\hat{V}^i) - X_k \right) \quad (9)$$

where \hat{V}^i is the current estimate of the blurred HR image, $(D_k \circ A_k)(\hat{V}^i)$ is the projection of the current estimate to

low resolution, and $(D_k \circ A_k)^{-1} \left((D_k \circ A_k)(\hat{V}^i) - X_k \right)$ is the *back-projection*. Irani et al.[10] also convolve the back-projection with a smoothing kernel as a form of regularization since the estimation problem is in general ill-posed (there are many \hat{V}^i that will project down to a pixel-distance neighbor of X_k). It can be shown[11] that \hat{V} is none other than the maximum likelihood estimate for V .

3.4 Example based

4 DEEP LEARNING ALGORITHMS

5 FUTURE RESEARCH

6 CONCLUSION

7 APPENDIX

TODO: work out diffraction circular aperture TODO: work-out poisson noise

ACKNOWLEDGMENTS

REFERENCES

- [1] M. Elad and Y. Hel-Or, "A fast super-resolution reconstruction algorithm for pure translational motion and common space-invariant blur," *IEEE Transactions on Image Processing*, vol. 10, no. 8, pp. 1187–1193, Aug. 2001. DOI: 10.1109/83.935034.
- [2] S.-C. Lin and C.-T. Chen, "Reconstructing Vehicle License Plate Image from Low Resolution Images using Nonuniform Interpolation Method," Tech. Rep. 1, p. 21.
- [3] M. S. Alam, J. G. Bogner, R. C. Hardie, and B. J. Yasuda, "Infrared Image Registration and High-Resolution Reconstruction Using Multiple Translationally Shifted Aliased Video Frames," Tech. Rep. 5, 2000.
- [4] S. Lertrattanapanich and N. K. Bose, "High resolution image formation from low resolution frames using delaunay triangulation," *IEEE Transactions on Image Processing*, vol. 11, no. 12, pp. 1427–1441, Dec. 2002. DOI: 10.1109/TIP.2002.806234.
- [5] R. Hardie, "A fast image super-resolution algorithm using an adaptive wiener filter," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2953–2964, Dec. 2007. DOI: 10.1109/TIP.2007.909416.
- [6] H. Takeda, S. Member, S. Farsiu, P. Milanfar, and S. Member, "Kernel Regression for Image Processing and Reconstruction," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 16, no. 2, p. 349, 2007. DOI: 10.1109/TIP.2006.888330. [Online]. Available: <http://www.soe.ucsc.edu/%20%7B%20~%20%7D%20htakeda..>
- [7] E. Nadaraya, "On Estimating Regression," *Theory of Probability & Its Applications*, vol. 9, no. 1, pp. 141–142, 1964. DOI: 10.1137/1109020. [Online]. Available: <https://doi.org/10.1137/1109020>.
- [8] M. Wand and M. Jones, *Kernel Smoothing*, ser. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1994, ISBN: 9780412552700. [Online]. Available: <https://books.google.com/books?id=GTOOi5yE008C>.
- [9] XiaoGuang Feng and P. Milanfar, "Multiscale principal components analysis for image local orientation estimation," in *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002.*, vol. 1, Nov. 2002, 478–482 vol.1. DOI: 10.1109/ACSSC.2002.1197228.
- [10] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Model and Image Processing*, vol. 53, pp. 231–239, 1991.
- [11] M. Elad and A. Feuer, "Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1646–1658, Dec. 1997. DOI: 10.1109/83.650118.