Super Resolution for Automated Target Recognition

Maksim Levental

- SR
  - MISR
    - Interpolation
      - Adaptive Filtering
    - Statistical
      - Back Projection
      - Markov Random Fields
      - Bilateral Total Variation
  - SISR
    - Manifold Learning
    - Compressed Sensing
    - Projection
    - Belief Networks

Fig. 1. Taxonomy of classical SR techniques

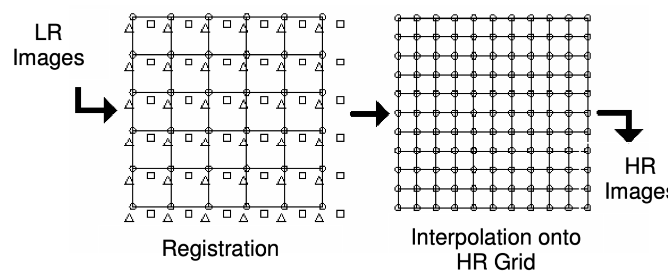

Fig. 2. LR image registration on an HR grid[2]

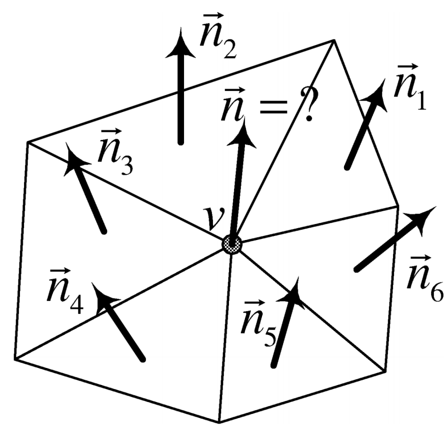

Fig. 3. Delaunay triangulation for fitting splines at LR pixels. v is an LR pixel. Note that v is at z equal to the pixel value.

***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 spatial[1] and time invariant. Suppose further that A_k is affine. Then $H := H_k$ commutes with A_k[1] and equation 3 becomes

$$
\begin{aligned}
X_k &= (D_k \circ A_k \circ H)(Y) + \epsilon && (1)\\
&= (D_k \circ A_k)(H(Y)) + \epsilon && (2)\\
&= (D_k \circ A_k)(Z) + \epsilon && (3)
\end{aligned}
$$

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

This naturally suggests interpolation in order to recover Z (since X_k, in this manifestation, is simply shifted samples of Z). 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 Z, 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 the adjacent pixels.

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). Lertrattanapanich et. al[4] base their algorithm on interpolants which

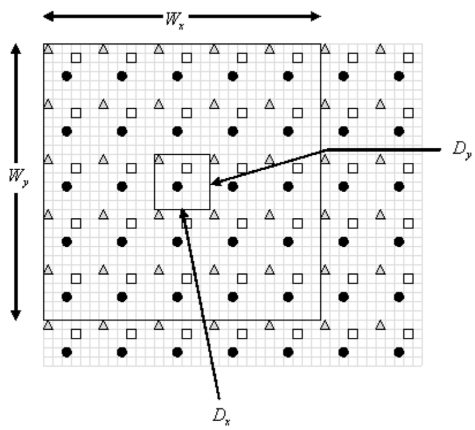

Fig. 4. Wiener filter super resolution estimation window of dimension $D_x \times D_y$ and observation window of dimension $W_x \times W_y$[5]

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[2] and compute the gradients (see figure 3) according to

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

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 pixels and the cross-correlation between LR pixels and interpolated pixels to estimate wiener filter weights. These weights are then used to average nearby pixel values. The algorithm operates on a sliding window called the 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 dimension W_x, W_y are an integer multiple of D_x, D_y (see figure 4). The weights $\boldsymbol{w}$ are defined as the solution to the minimum mean squared error filter problem, i.e. the finite impulse response (FIR) wiener filter:

$$\boldsymbol{w} = R^{-1}\boldsymbol{p} \tag{4}$$

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

$$C_1(r) = \sigma_d^2 \rho^r * G(r) \tag{5}$$

where 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 $\boldsymbol{p}$ is constructed by sampling from

$$C_2(r) = \sigma_d^2 \rho^r * G(r) * G(-r) \tag{6}$$

where here $r = r(m, n)$ 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 $\boldsymbol{p}$ is an $N \times 1$ column vector and unique 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.

Takeda et al.[6] approach the problem by leveraging recent advances (at the time) in kernel regression; they

3.3 Estimation

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

[2]An 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.

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. Bognar, 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. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, Feb. 2007. DOI: 10.1109/TIP.2006.888330.