Image Super-Resolution via Least-Squares and Robust LOESS

Shao-Chun Lu(呂紹群), Chih-Chun Ho (何智鈞), ***Department of Mathematics, National Taiwan Normal University*** (41140104S@gapps.ntnu.edu.tw)
Advisor: Prof. Shih-Feng Shieh(謝世峰), Prof. Je-Chiang Tsai(蔡志強)

Abstract

We propose a super-resolution pipeline that converts a low-resolution image into a sharper reconstruction. The workflow proceeds in three stages. First, missing pixels are inferred via local quadratic least-squares up-sampling. Second, a two-pass robust LOESS smoothes the image. Third, the high-frequency residual between the up-sampled image and its smoothed counterpart is reintroduced with an optimal α, yielding a crisp result. Across diverse image, our method consistently outperforms conventional bilinear up-sampling in both perceptual quality and quantitative error metrics.

Methodology

Super-resolving a down-sampled image requires interpolation of missing samples and suppression of noise. We split the task into a sequence of local polynomial regression, low-pass smoothing, and selective enhancement, each stage operating on closed-form equations.

Fig. 1: original LR patch;
Fig. 2: after least-squares up sampling;
Fig. 3: after LOESS smoothing;
Fig. 4: after robust LOESS and adaptive boost).

Step 1 – Local Least-Squares Upsampling

For every empty high-resolution pixel we gather its 5 × 5 neighborhood, center the target to (0, 0), and fit the quadratic surface:

$$\hat{z}(x,y) = a\,x^2 + b\,xy + c\,y^2 + d\,x + e\,y + f$$

The coefficient vector follows the equation:

$$\mathrm{a} = \begin{bmatrix} a & b & c & d & e & f \end{bmatrix}^\top = (A^\top A)^{-1} A^\top \mathbf{z}$$

with

$$\mathrm{A} = \begin{bmatrix} x^2 & xy & y^2 & x & y & 1 \end{bmatrix}$$

Step 2 – LOESS Smoothing

For every pixel $(\mathrm{x}_0,\ y_0)$ we gather all neighbors $(\mathrm{x}_i,\ y_i)$ whose Euclidean distance:

$$\mathrm{r}_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}$$

Each neighbor receives a weigh($\mathrm{R}=5$):

$$\mathrm{w}_i = \left(1 - (r_i/R)^3\right)^3, \qquad 0 \le r_i \le R$$

X is an $\mathrm{n} \times (p+1)$ matrix with entries
p = polynomial degree.
n = # of the pixels in neighbor R.
z = values corresponding to each x_i
$\mathrm{W} = \mathrm{diag}(w_i)$

The weighted least-squares coefficients are obtained in closed form:

$$\boldsymbol{\beta}(x_0, y_0) = (X^\top W X)^{-1} X^\top W\,\mathbf{z}$$

The LOESS surface at the center pixel is

$$\mathrm{I}_{\mathrm{smooth}}(x_0, y_0) = \beta_0$$

Step 3 – Robust LOESS High-Frequency Boost

with the least squared up sample: $\mathrm{I}_{\mathrm{LSM}}$ yields the high-frequency layer:

$$\mathrm{D} = \mathrm{I}_{\mathrm{LSM}} - I_{\mathrm{smooth}}$$

we exploit the source reference X_{HR}
Let

$$\mathrm{E} = \mathrm{I}_{\mathrm{LSM}} - X_{\mathrm{HR}}$$

be the pre-boost reconstruction error. Minimizing the mean-squared error of the boosted image

$$\mathrm{I}_{\mathrm{final}} = I_{\mathrm{LSM}} + \alpha D$$

the scalar gain α gives the solution

$$\hat{\alpha} = -\frac{\langle D, E\rangle}{\langle D, D\rangle + \varepsilon}$$

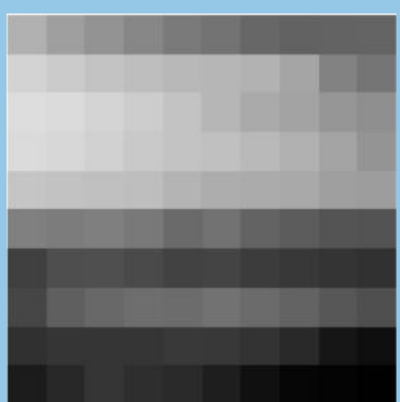


Fig.1

LSM Upsample →

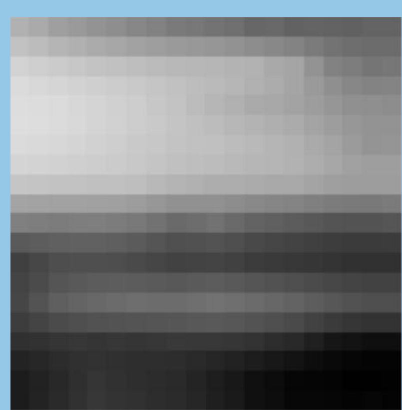


Fig.2

LOESS Smooth →

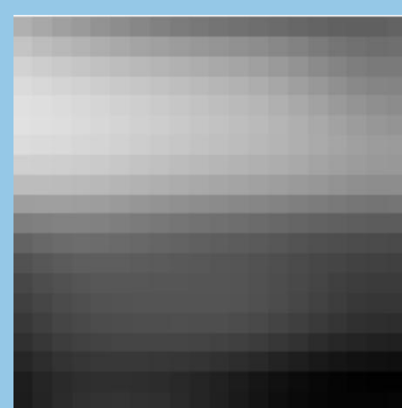


Fig.3

LOESS Boost →

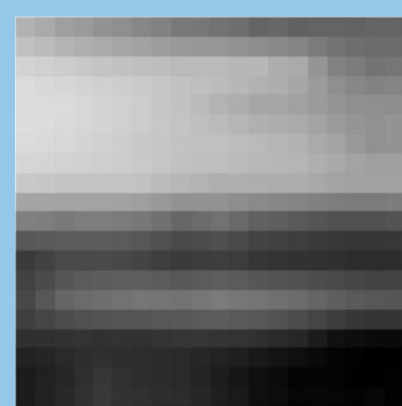
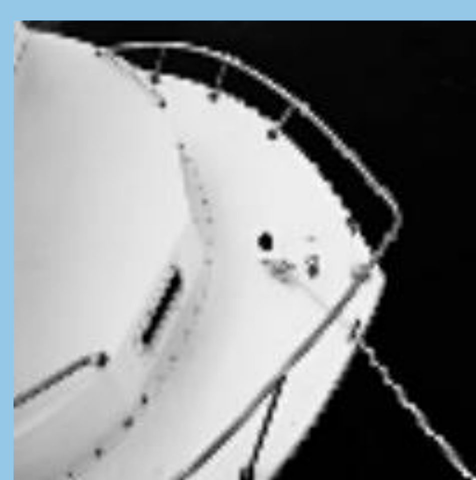


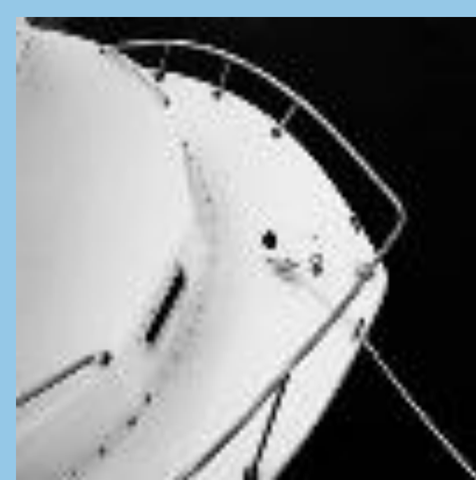
Fig.4

Results and discussion

→ α = 0.11
SSIM = 0.936
PSNR = 29.27



→ α = 0.05
SSIM = 0.933
PSNR = 25.06



As we can see in this result, the left is a $100{\times}100$ input, after our method we will create a $200{\times}200$ image on the right. The edges of the building and windows is not as pixelated. This method outperforms the bilinear method which is the default method on MATLAB

We can observe that the white line in this image is sharper and clearer compared with the input image. This highlights one of the key strengths of our method which is smoothing the overly pixelated parts while keeping its sharpness in the image. This also outperforms the Bilinear method.

Future work

As the results show, we have not found the optimal method for super-resolution imaging. We speculate that revising and tuning our least-square-method will give better results. This method also relies on the source itself to find the α value, we intend to find a way to calculate α without the source image.

References

1 Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. Journal of the American Statistical Association, 74(368), 829–836. https://doi.org/10.1080/01621459.1979.10481038
2 hfreepik.com, https://www.freepik.com/free-photo/low-angle-plain-concrete-structure-city-with-copy-space_12065853.htm
3 unsplash.com, https://unsplash.com/photos/a-close-up-of-a-white-boat-in-the-water-5pIzaPeAukU