Single Image Interpolation Exploiting Semi-Local Similarity

Lantao Yu Michael Orchard

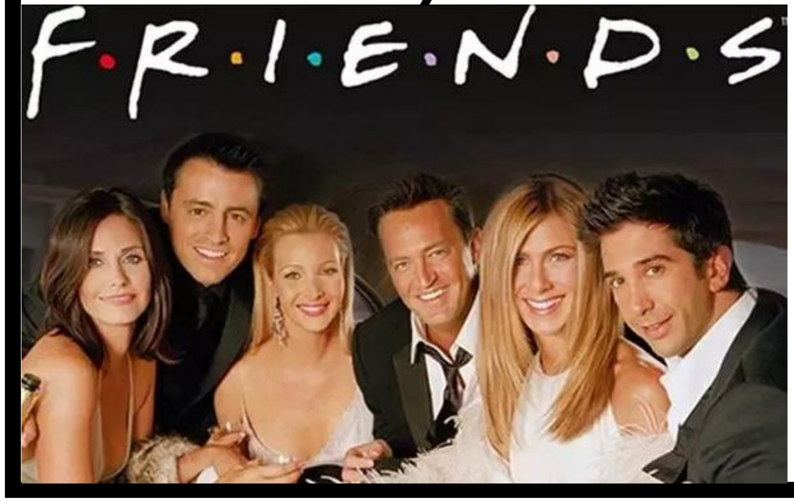
Department of Electrical and Computer Engineering
Rice University

RICE

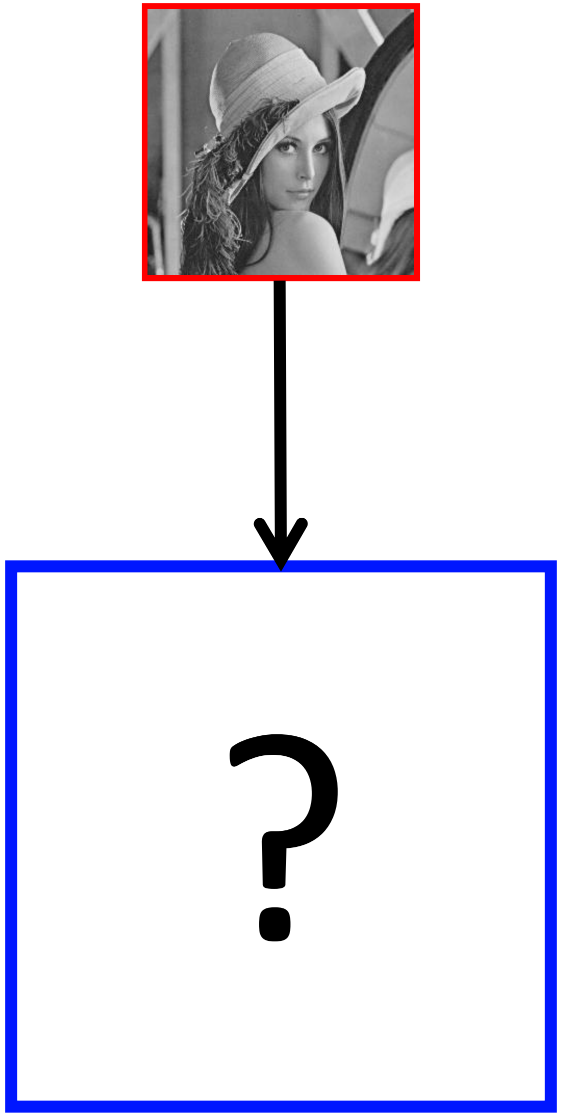
Ubiquitous Need

iMac (Late 2013) : 1920×1080

Friends (debut 1994) : 720×480

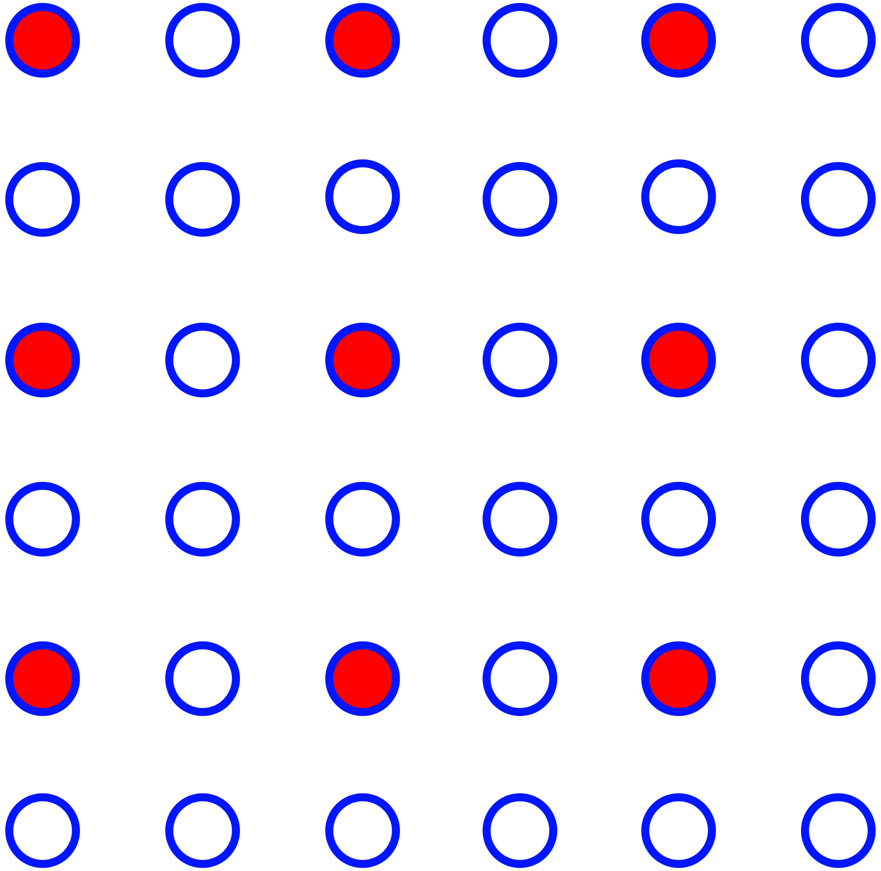


The Interpolation Problem



Low-Resolution (LR) image

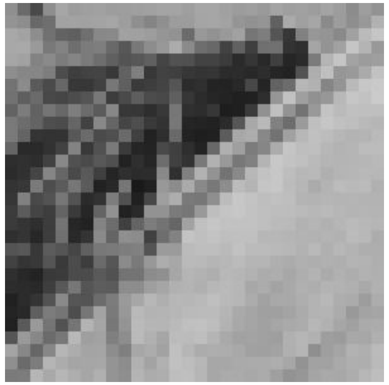
High-Resolution (HR) image



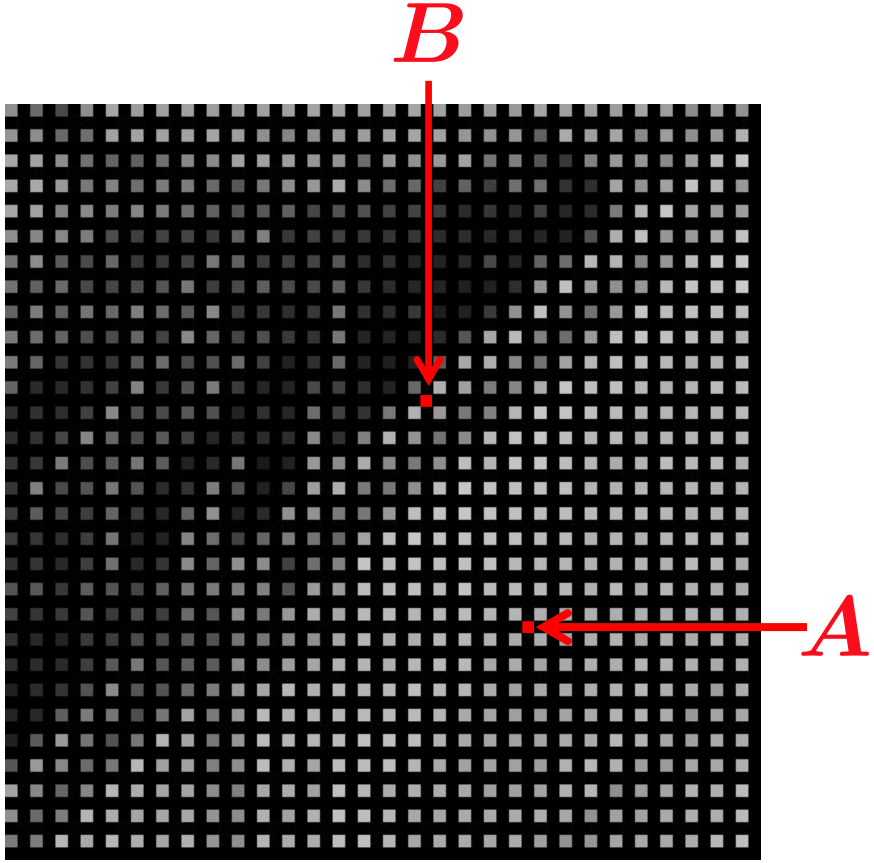
🔴 Measured Pixels: from LR image

○ Missing Pixels: to be estimated

Challenge



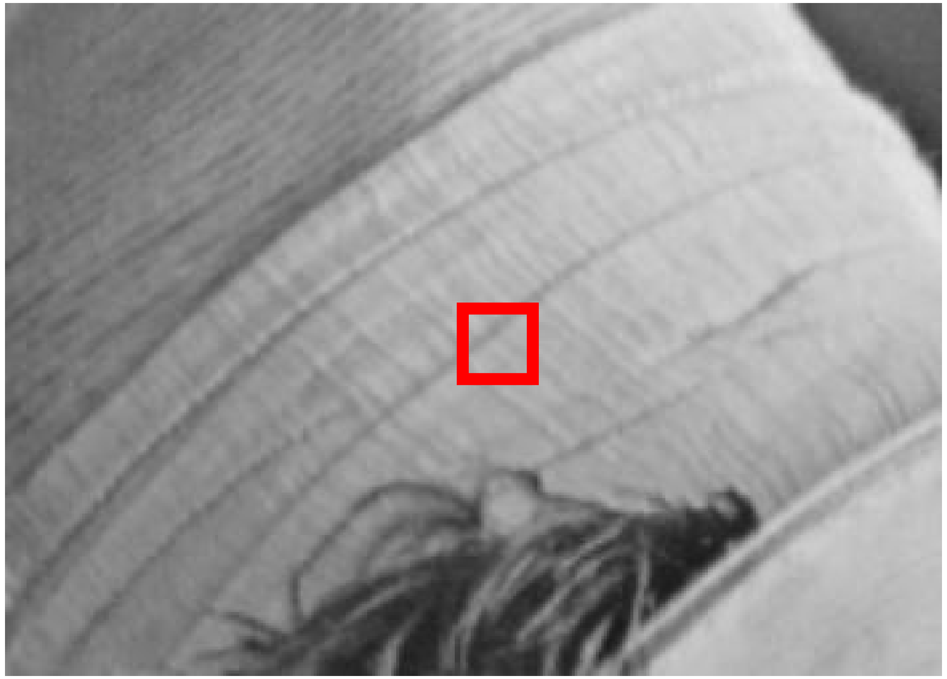
LR image



Upsampled-by-2
LR image

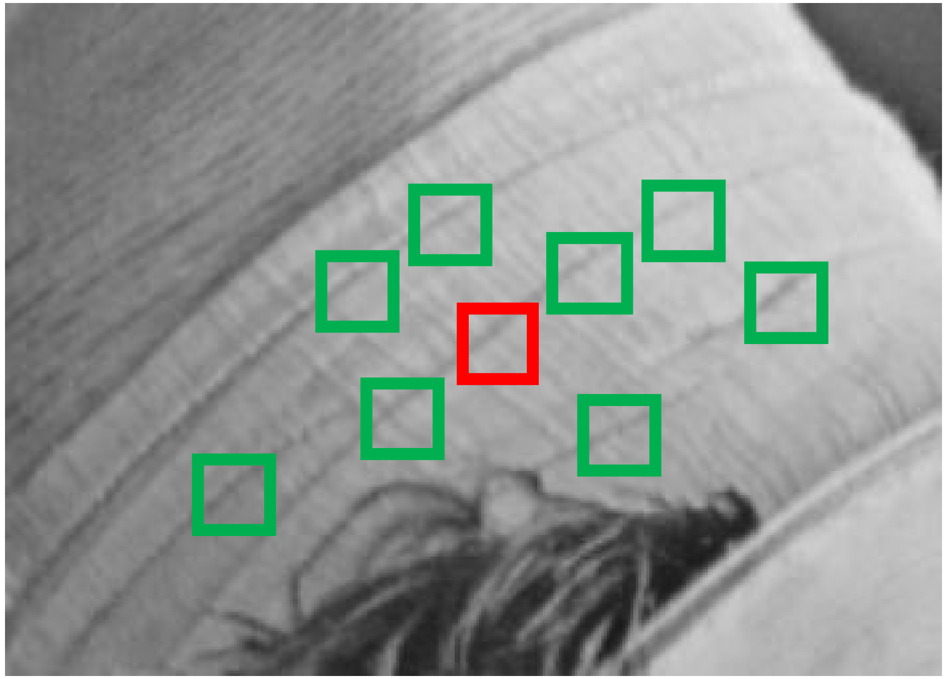
HR Image

Semi-Local Similarity



A Patch (the red block of pixels) and its Neighboring Similar Patches (the green blocks of pixels) in *Lena.*

Semi-Local Similarity



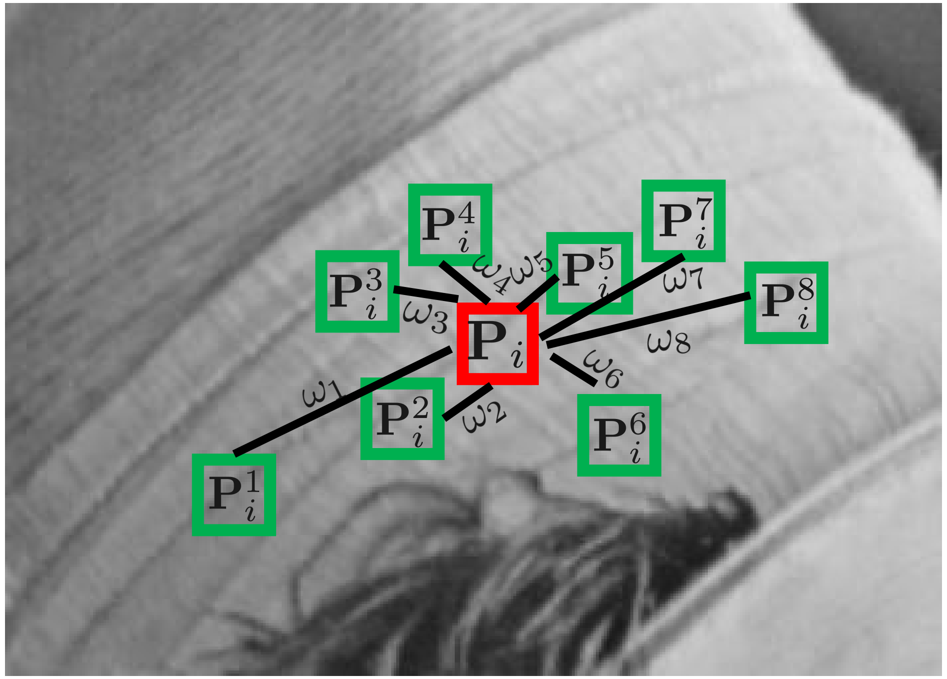
A Patch (the red block of pixels) and its Neighboring Similar Patches (the green blocks of pixels) in *Lena.*

Semi-Local Similarity

$$\mathbf{P}_i \approx \sum_j \omega_j \mathbf{P}_i^j$$

An individual patch in a natural image

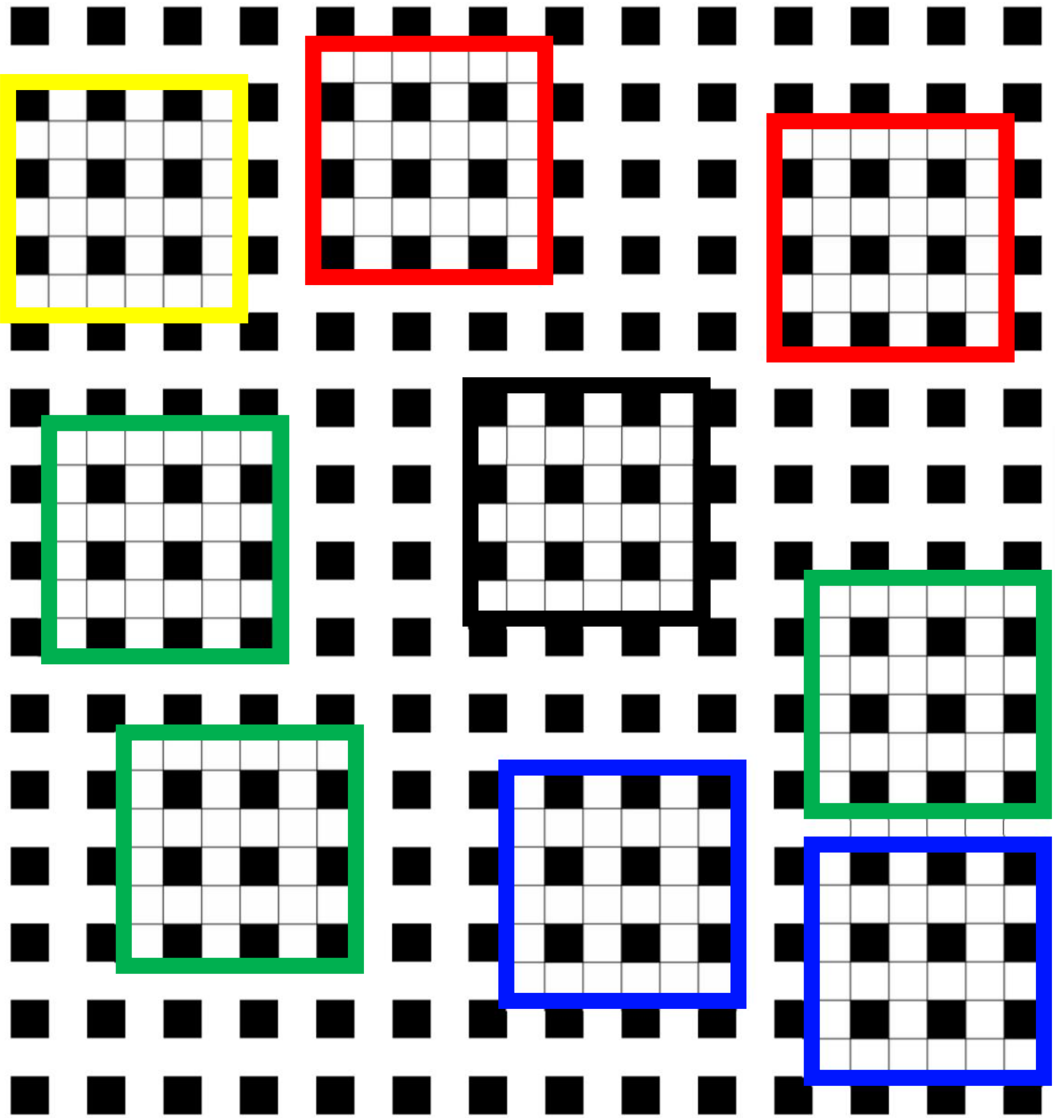
Similar patches searched in $\mathbf{P}_i$'s spatial neighborhood



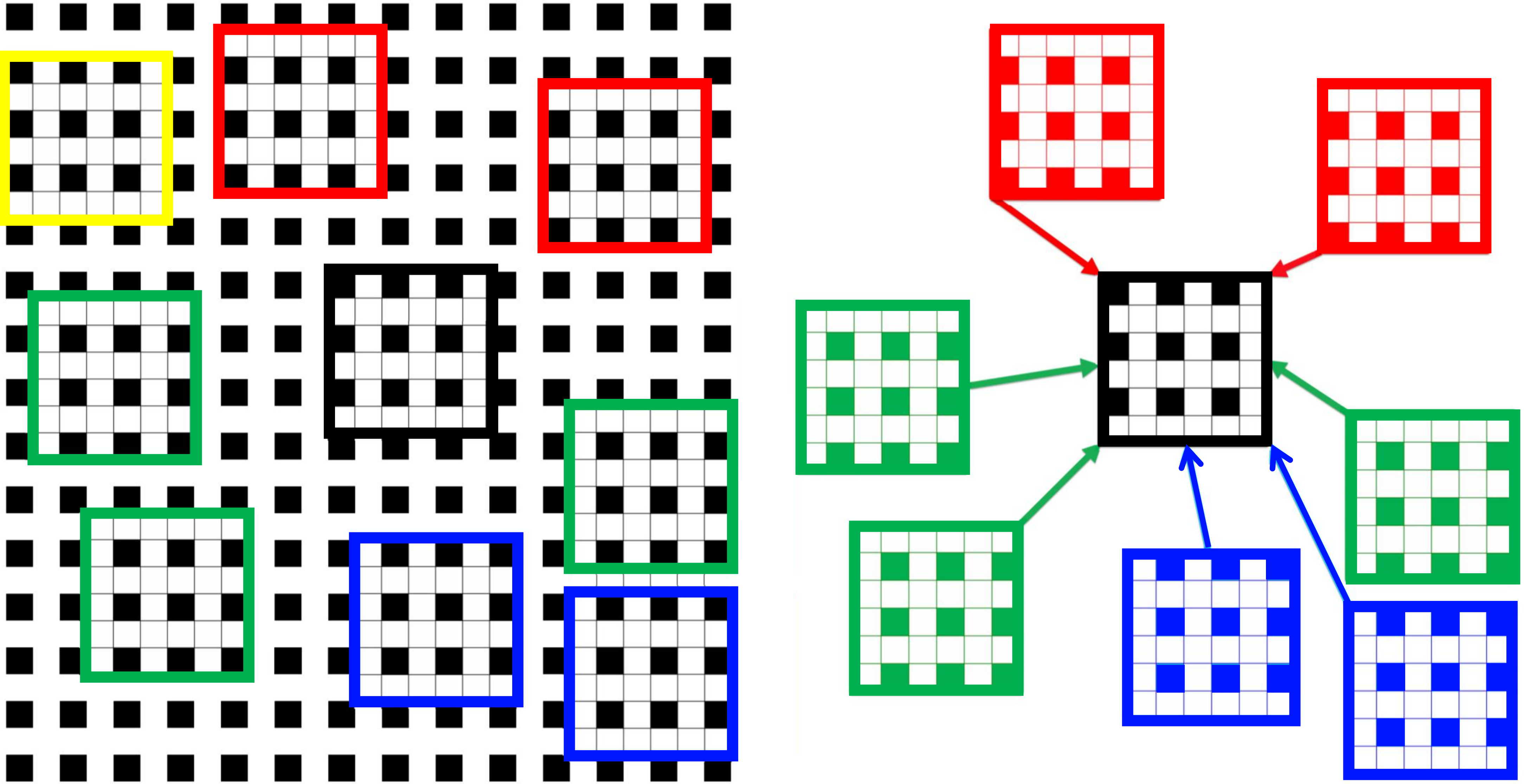
A Patch (the red block of pixels) and its Neighboring Similar Patches (the green blocks of pixels) in *Lena.*

Buades et al. CVPR 2005
Dong et al. TIP 2013

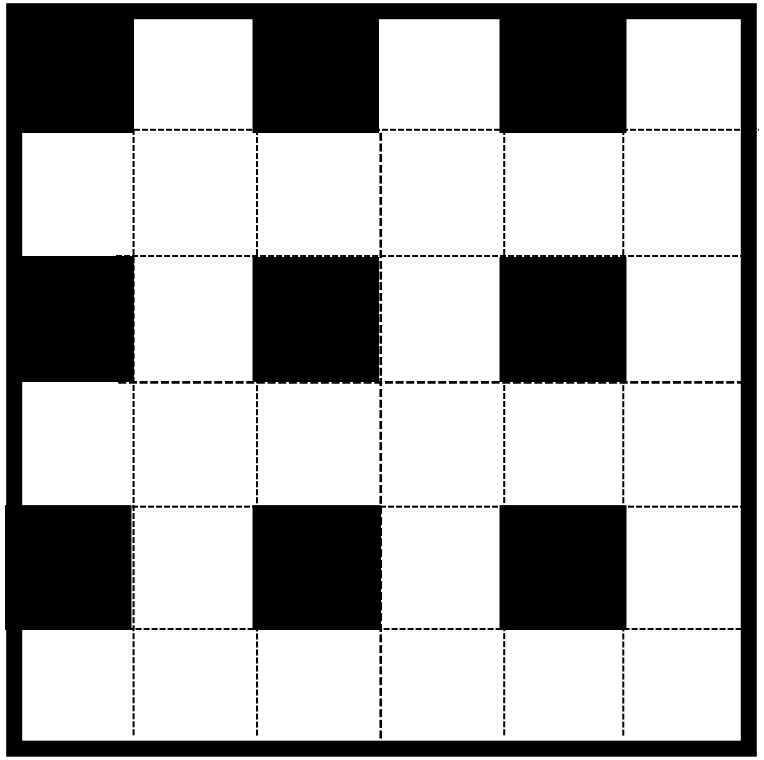
Exploit Semi-Local Similarity

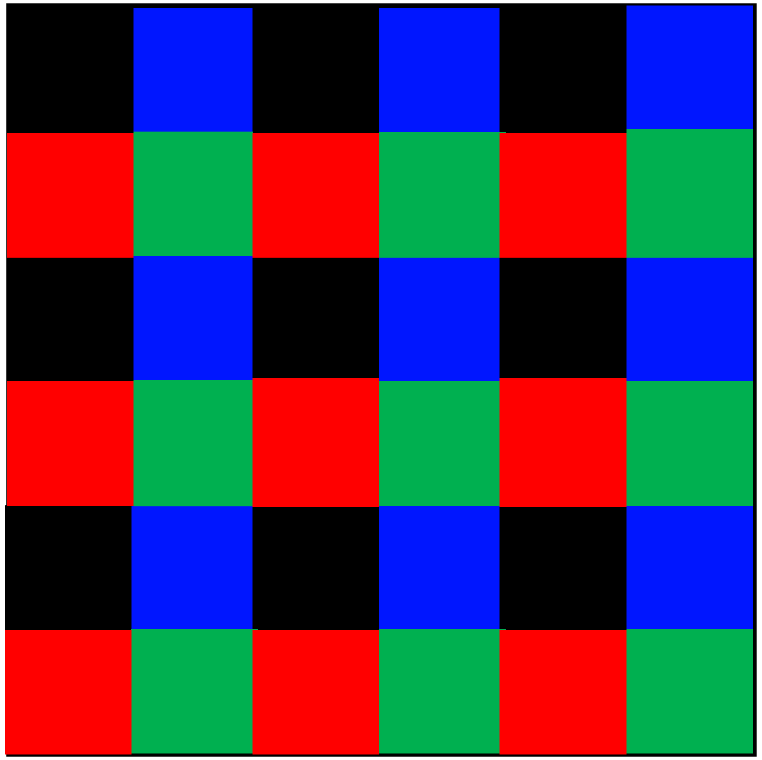


Exploit Semi-Local Similarity



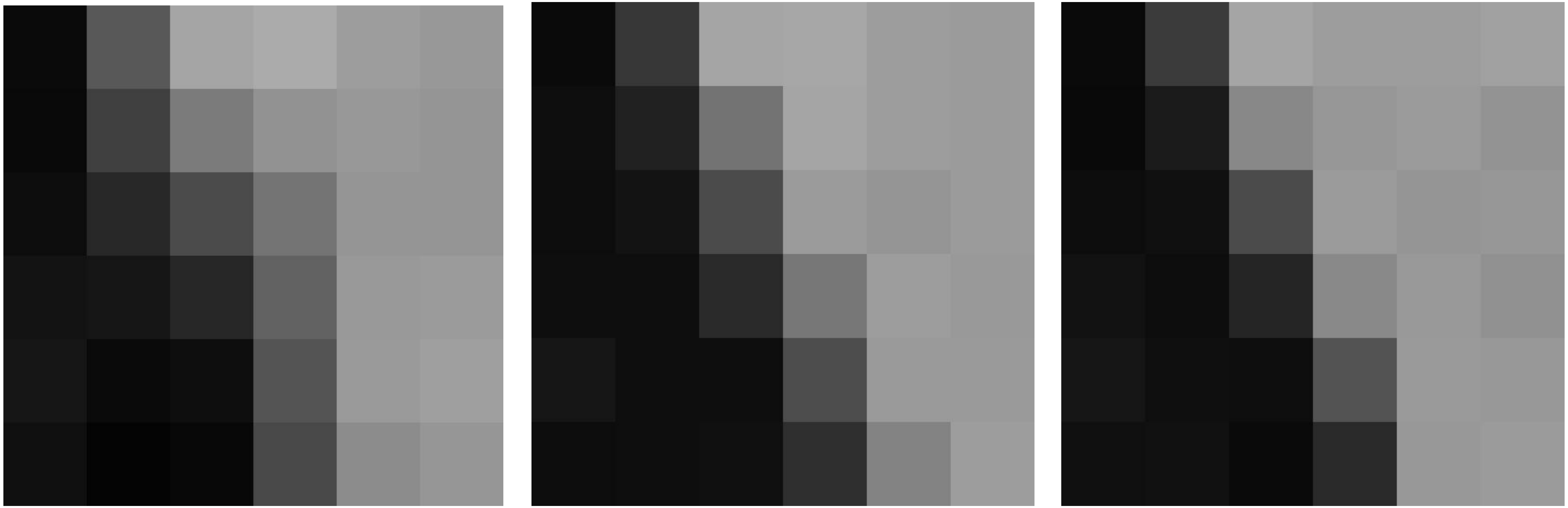
Exploit Semi-Local Similarity





Sun et al. DSP 2015

Exploit Semi-Local Similarity



(a) initial estimate (b) after first iteration (c) ground-truth

Scheme of Each Iteration

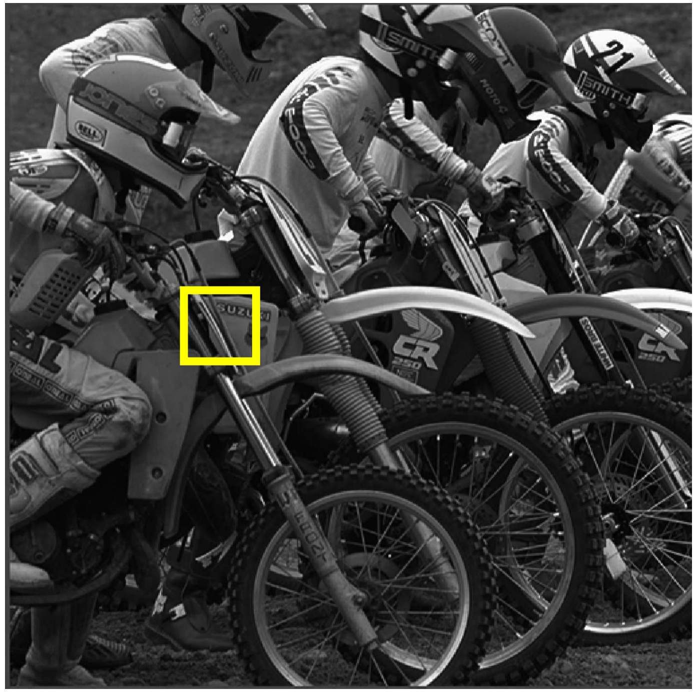
- □ Decompose an image into overlapping patches
- □ Update each patch:
  - ❖ Identify the positions of similar patches
  - ❖ Compute the weights of similar patches
- □ Average the contribution of overlapping patches to each missing pixel

Unique Features

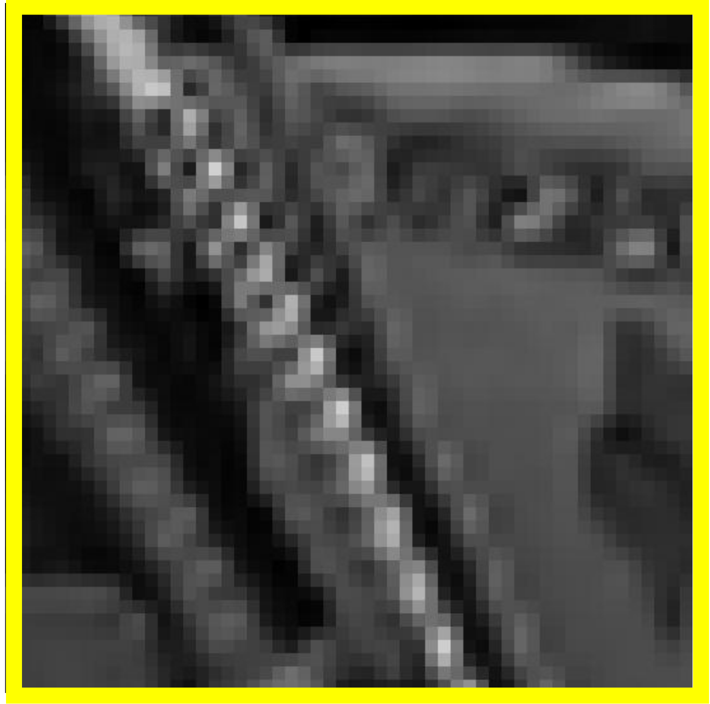
□ Robust initialization of the positions of similar patches

□ Regularization of the weights of similar patches

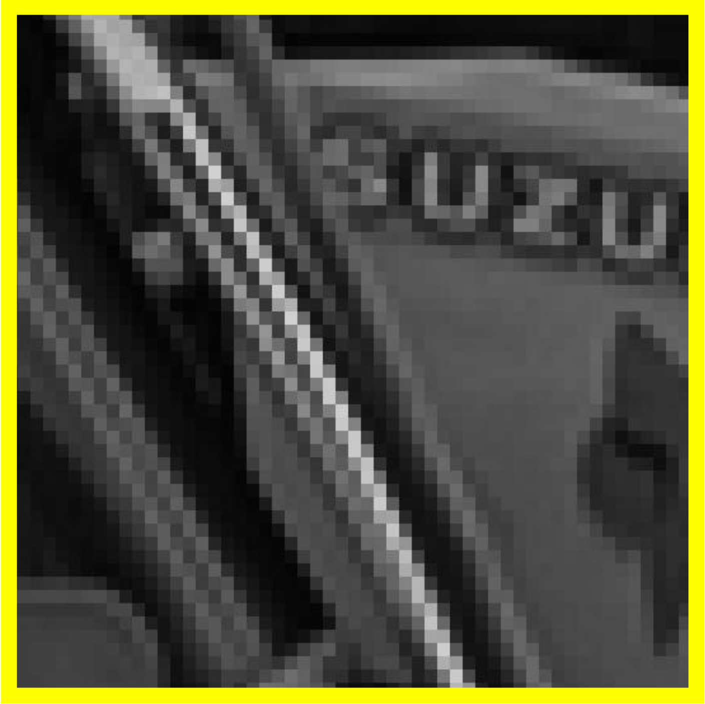
Robust Position Initialization



HR Image and a window of interest

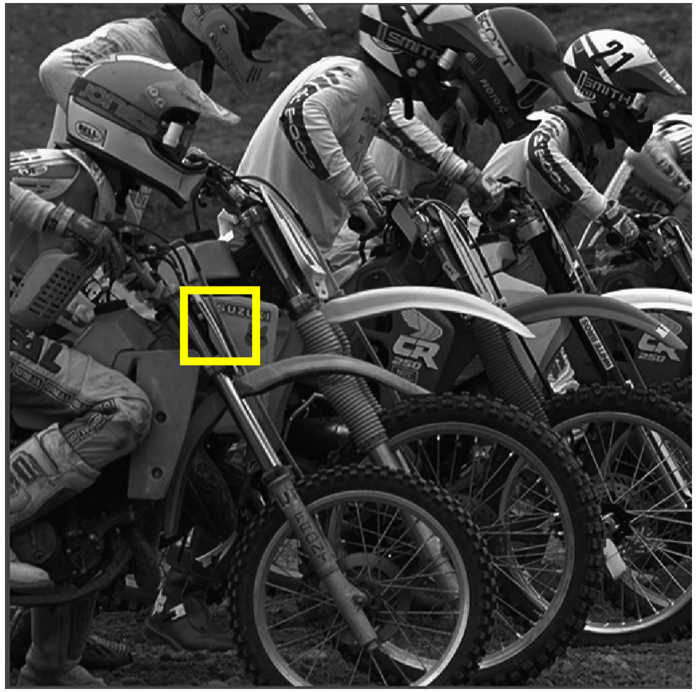


Bicubic Initial Estimate

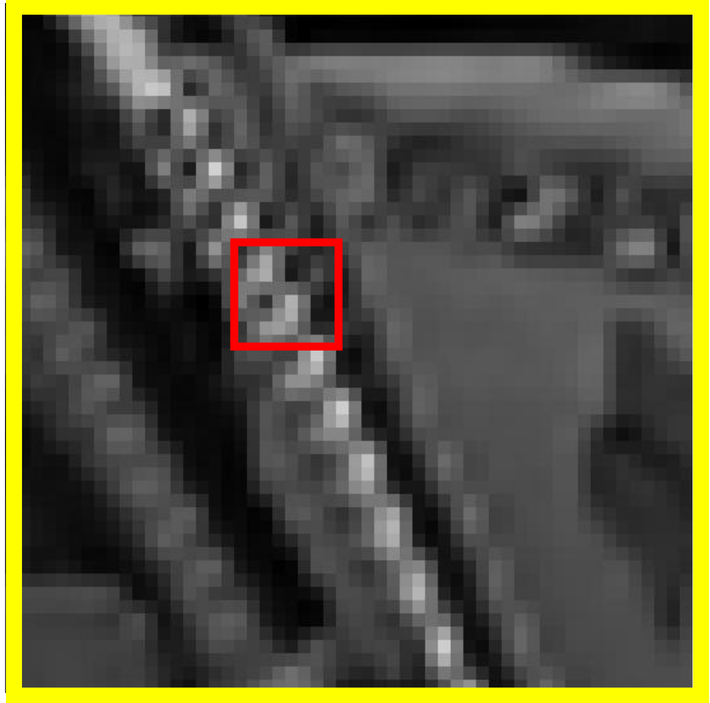


HR image

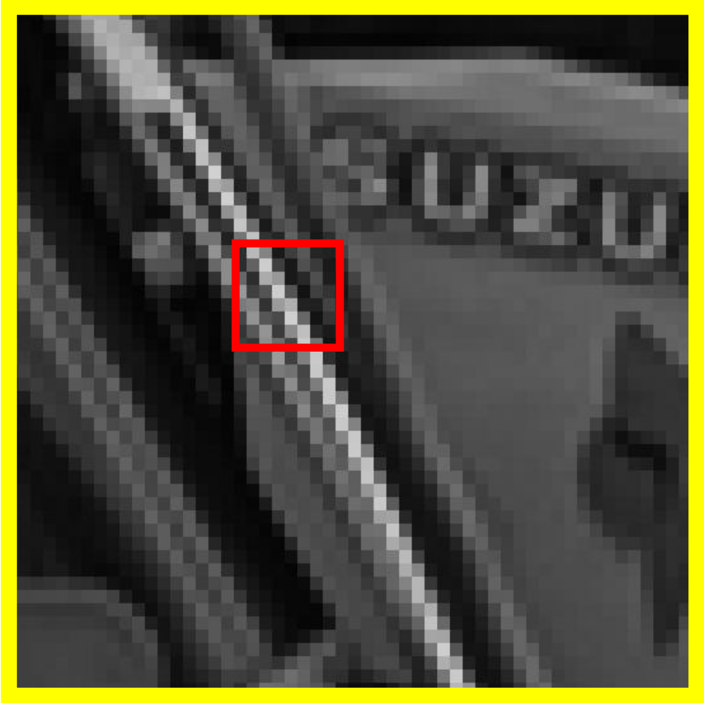
Robust Position Initialization



HR Image and a window of interest

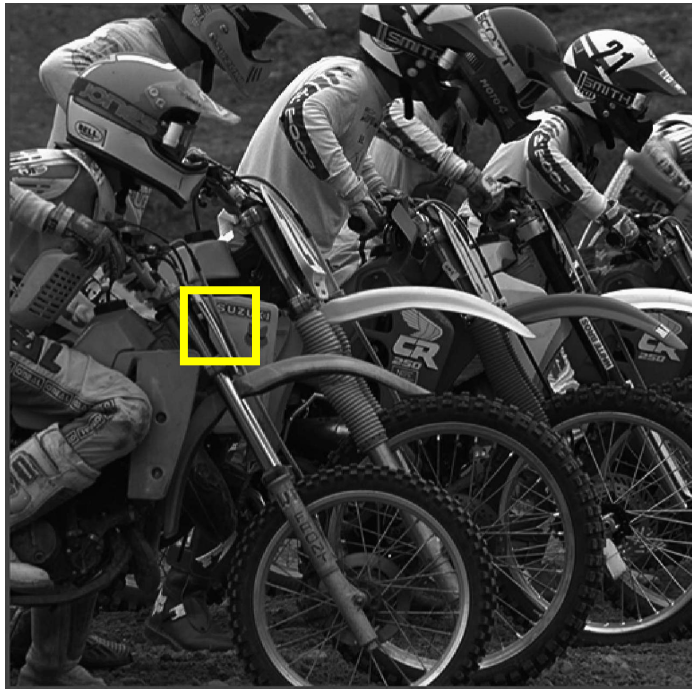


Bicubic Initial Estimate
target patch

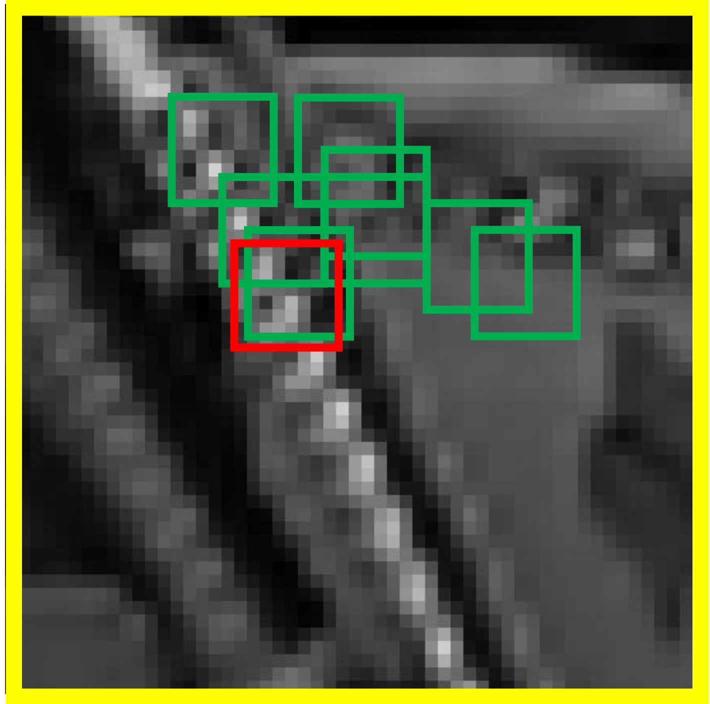


HR image
target patch

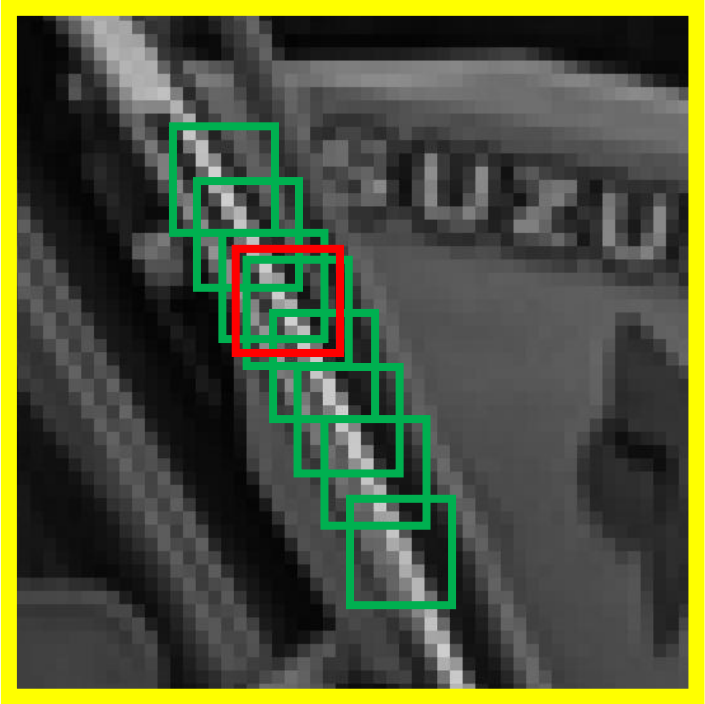
Robust Position Initialization



HR Image and a window of interest

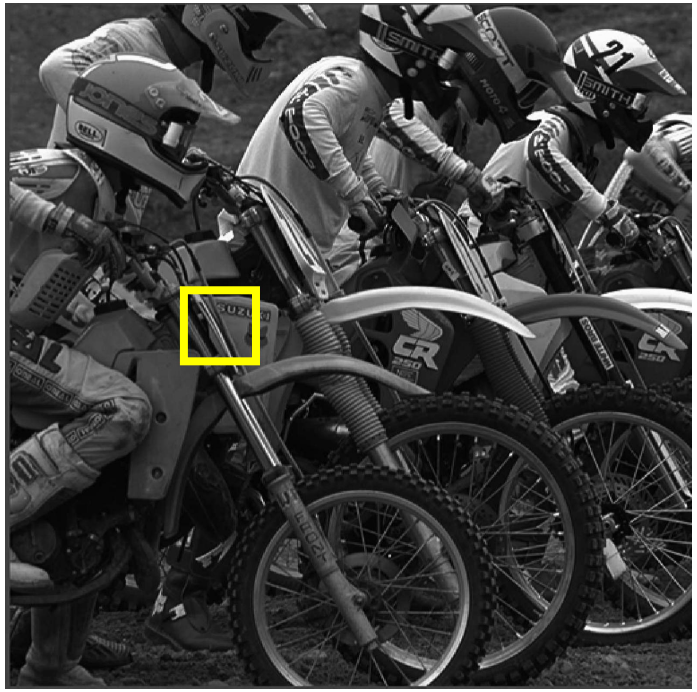


Bicubic Initial Estimate
target patch and
similar patches

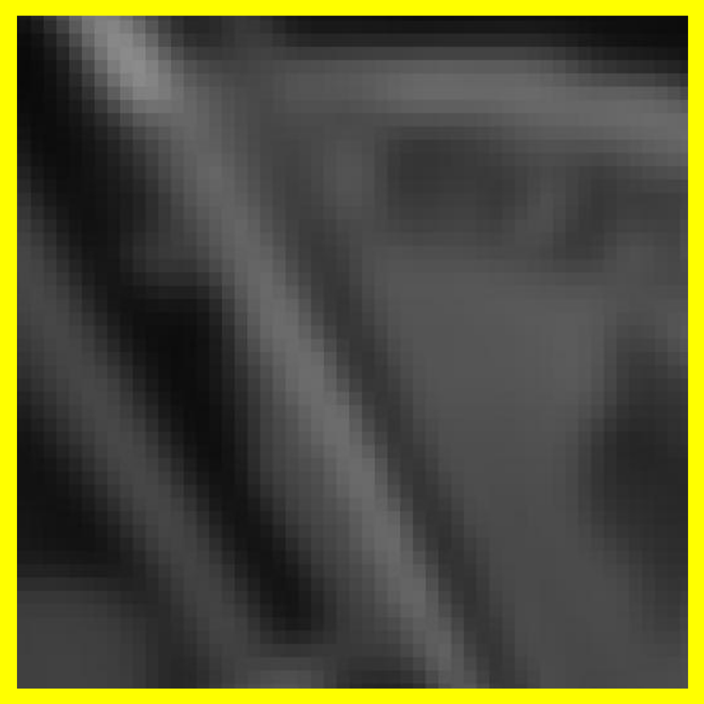


HR image
target patch and
similar patches

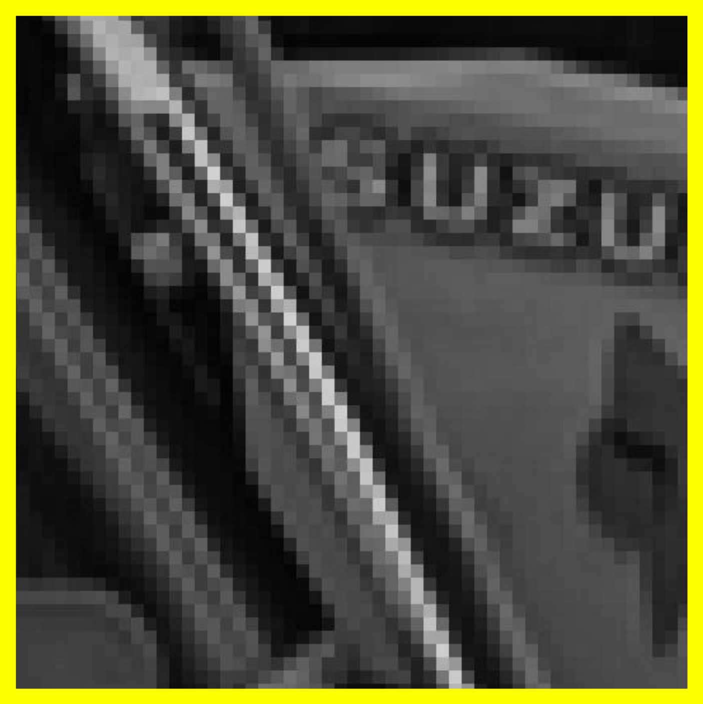
Robust Position Initialization



HR Image and a window of interest

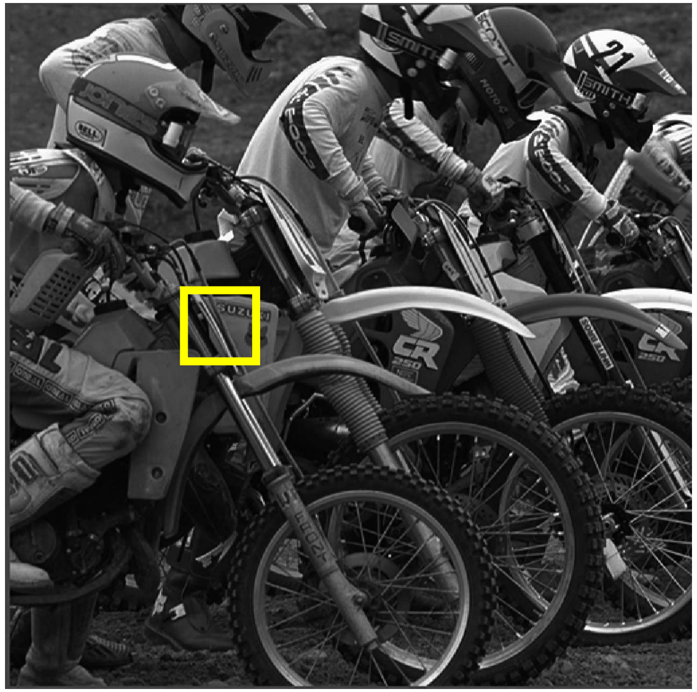


Guide Image

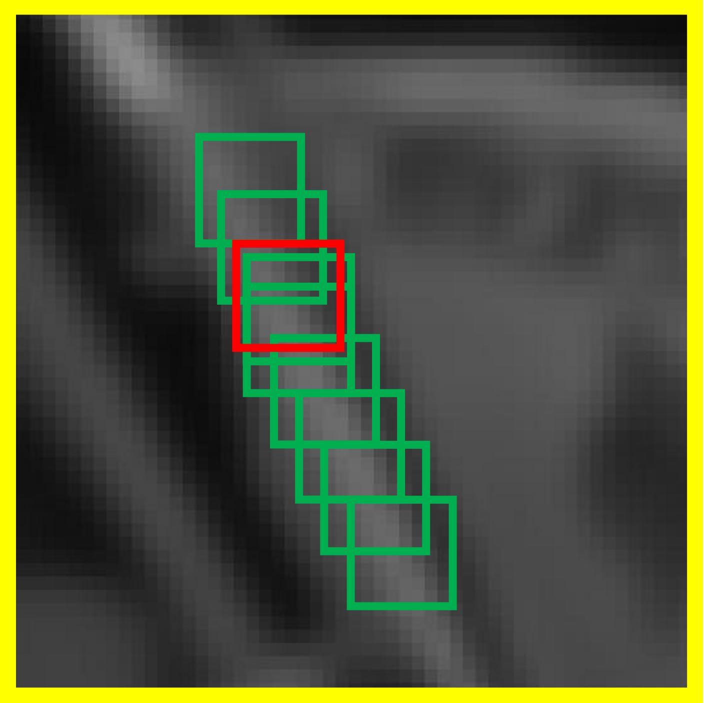


HR image

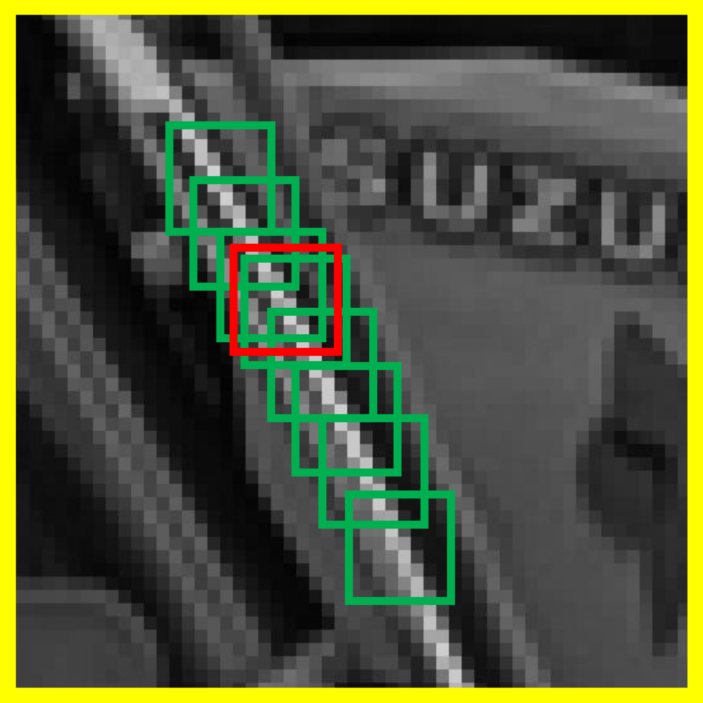
Robust Position Initialization



HR Image and a
window of interest

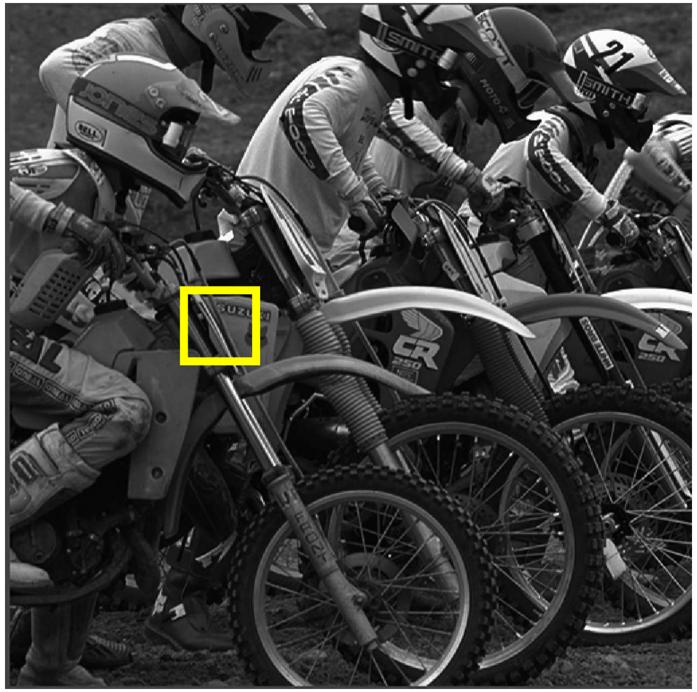


Guide Image
target patch and
similar patches

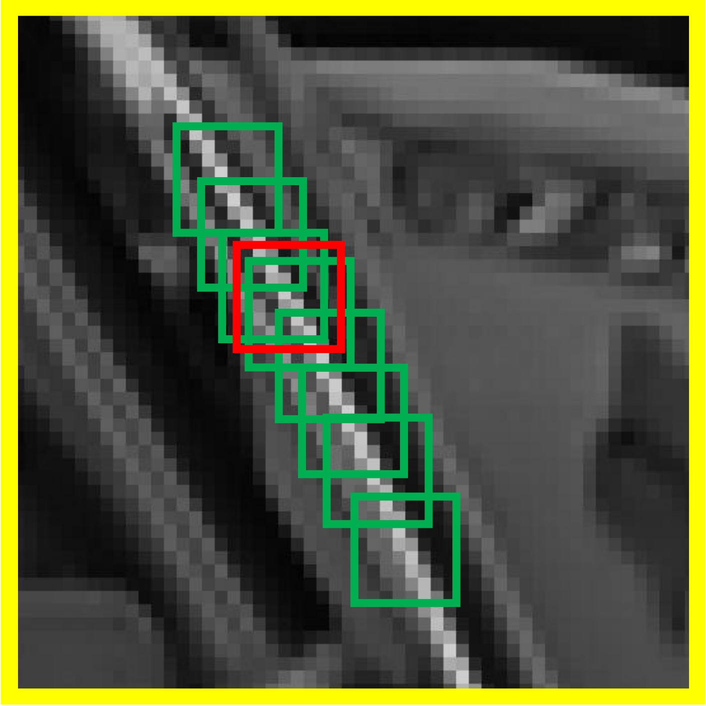


HR image
target patch and
similar patches

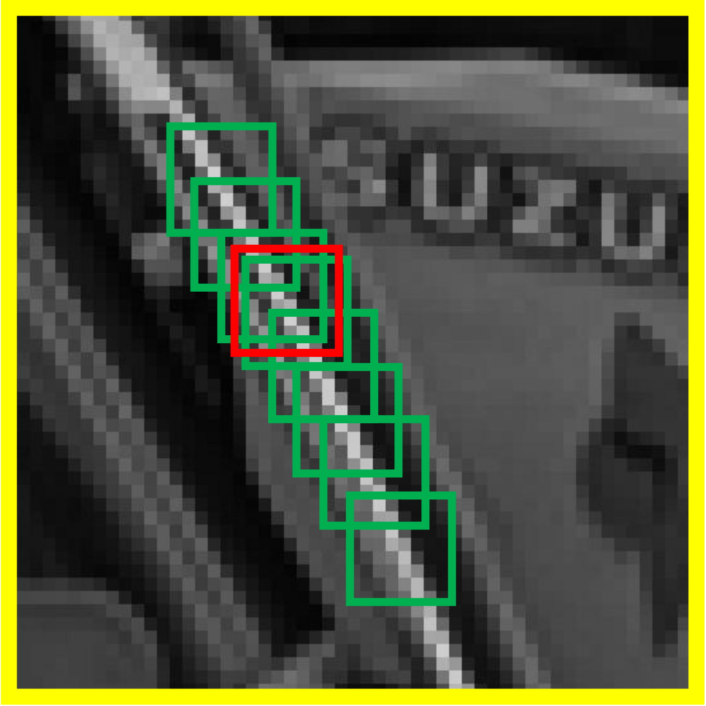
Robust Position Initialization



HR Image and a window of interest



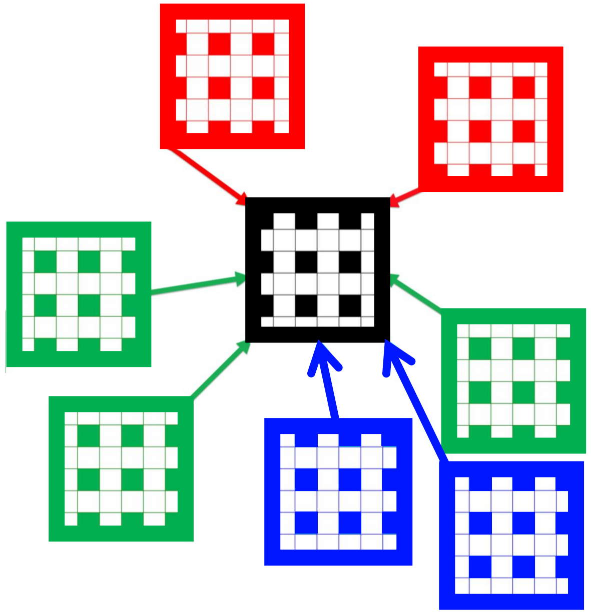
Input Image in Last Iter target patch and similar patches



HR image target patch and similar patches

Regularized Weights

$$\omega = \arg\min \left[\underbrace{\left\|\boldsymbol{Q}_i \omega - \boldsymbol{p}_i\right\|_2^2}_{\text{Data Fidelity Term}} + \underbrace{\lambda \omega^T \boldsymbol{\Sigma} \omega}_{\text{Penalty Term}} \right]$$



Testset

USC-SIPI Database
Berkeley Segmentation Database

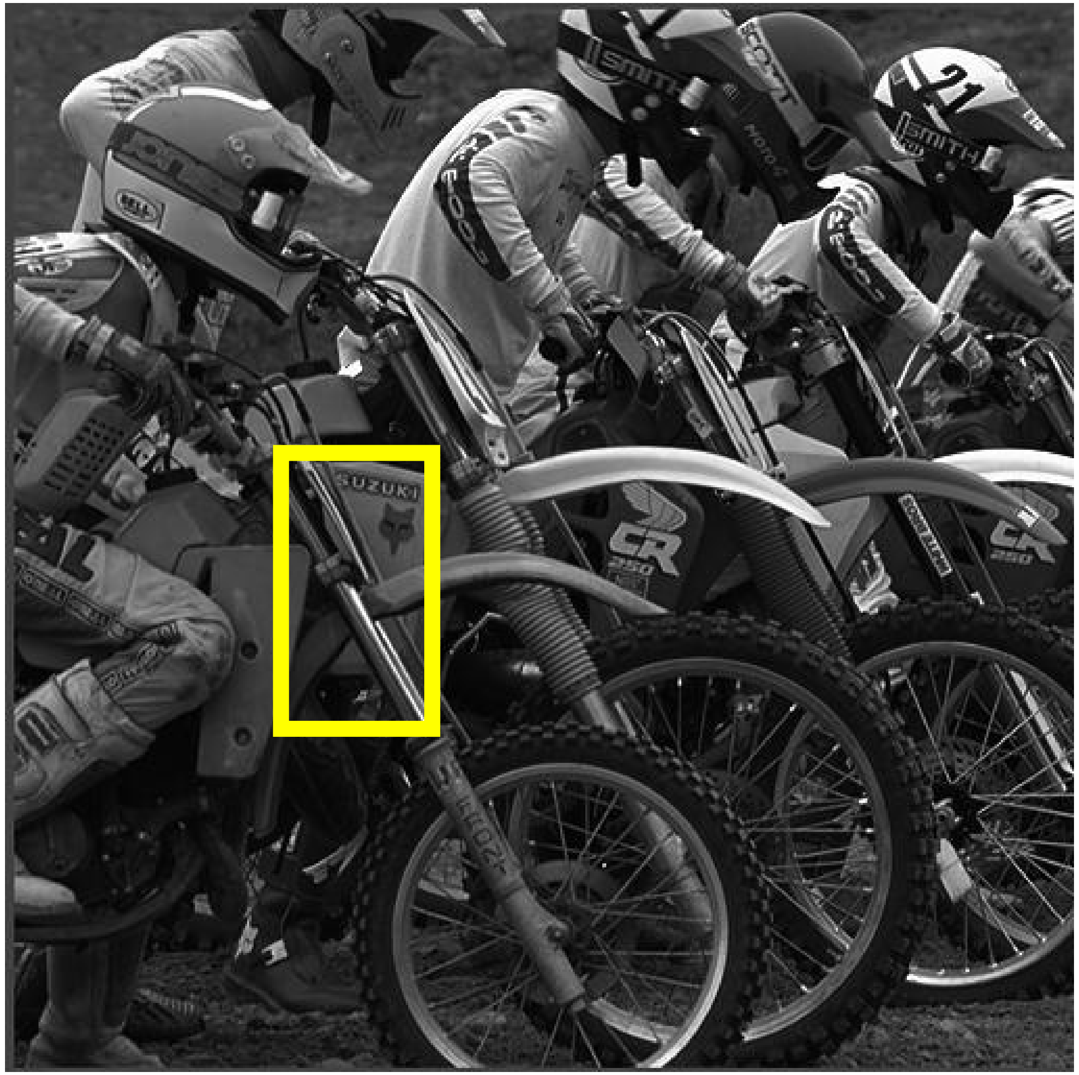
Quantitative Comparison

Table 1: Comparison of Average PSNR (in decibels) of interpolated images.

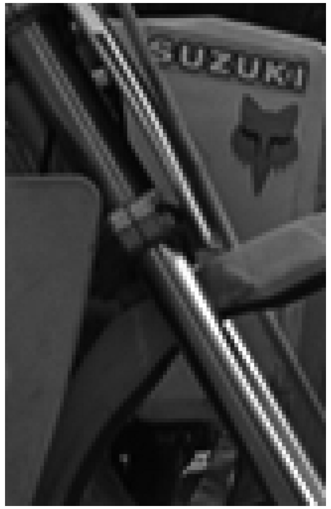
Method	X2	X3
NEDI [Li & Orchard, 2001]	30.25	/
SAI [Zhang & Wu, 2008]	30.80	/
SME [Mallat & Yu, 2010]	30.74	/
RLLR [Liu et al., 2011]	30.80	/
NARM [Dong et al., 2013]	31.20	27.29
ANSM [Romano et al., 2014]	31.32	27.54
NLPC [Sun et al., 2015]	31.31	27.64
NGSDG [Zhu et al., 2016]	30.76	/
Proposed	**31.64**	**27.86**

*$\mathrm{PSNR} = 20\log_{10}\left(\frac{255}{\sqrt{\mathrm{MSE}}}\right)$

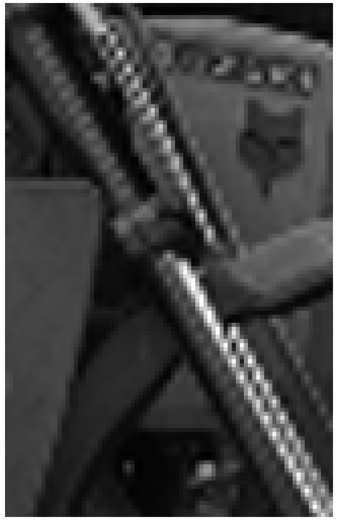
Visual Comparison (X2)



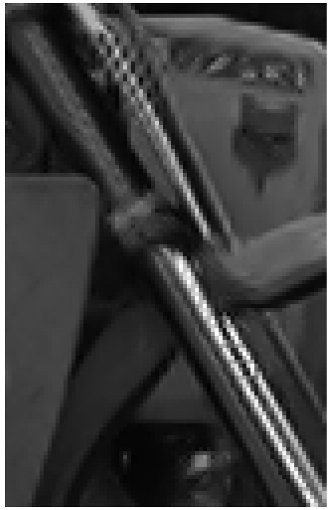
Visual Comparison (X2)



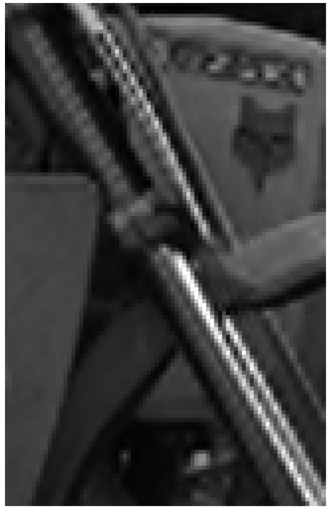
Original



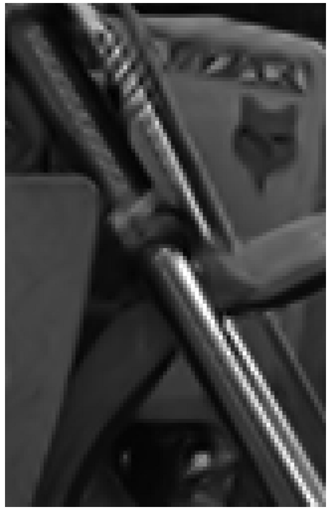
Bicubic



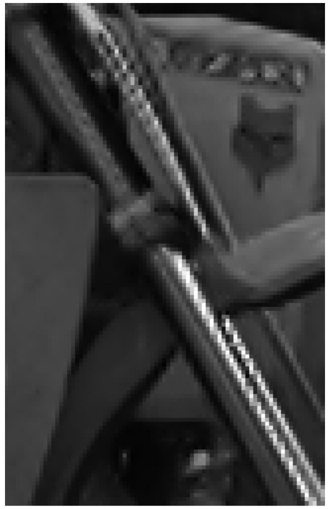
NEDI



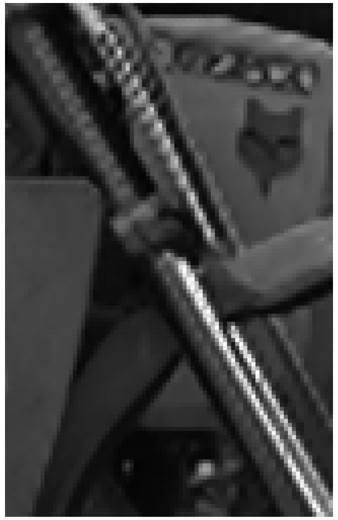
SME



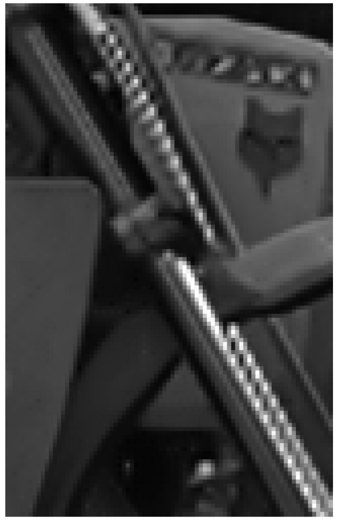
SAI



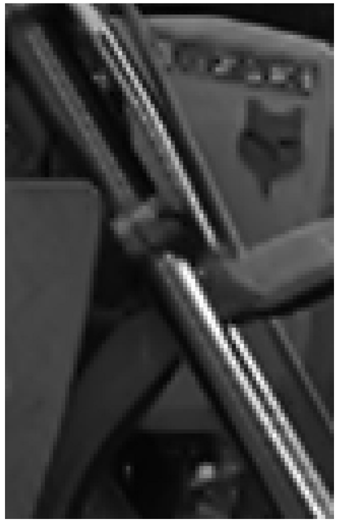
RLLR



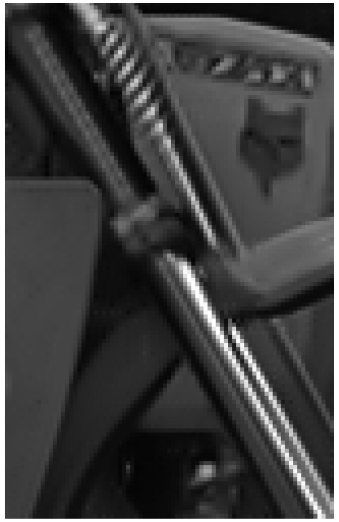
NGSDG



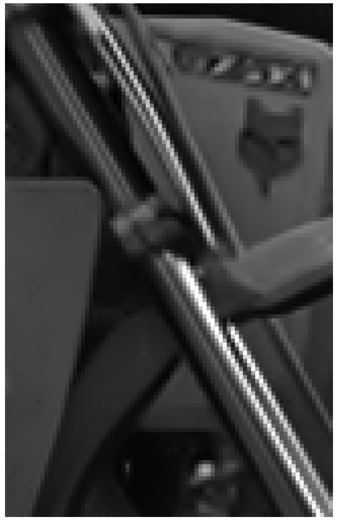
NARM



ANSM



NLPC



Proposed

Runtime Comparison (X2)

Table 2: Comparison of runtimes (in seconds) of interpolating a 128X128 LR image to a 256X256 HR image. Platform: Intel 2.6G Hz 18-Core Intel i9 processor, Matlab R2018b, C++ (GCC 7.3.0), Eigen 3, OpenMP.

Method	Programming Details	Runtime/second
NARM [Dong et al., 2013]	Matlab, Non-Parallel	43.4 ∼ 138.9
ANSM [Romano et al., 2014]	Matlab, C/C++ MEX	1265.0
Proposed	Matlab, Non-Parallel	155.6
	Matlab, Parallel	13.3
	C++, Parallel	3.6

Conclusion

□ Novel Way to Improve Exploitation of Semi-Local Similarity

□ State-of-the-art PSNR

□ Simple, Parallerizable Algorithm