A STUDY OF MODERN IMAGE UPSCALING METHODS

REVIEW III REPORT

Submitted by

R Aditya Sai (18BCE0034)

Prepared For

IMAGE PROCESSING (CSE4019) –REVIEW 3

Submitted To

Dr. Swathi J.N

Assistant Professor (Sr)

School of Computer Science and Engineering

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

ACKNOWLEDGEMENT

This work is a part of the **IMAGE PROCESSING** course project component. I would like to thank VIT and Asst. Prof. Dr. SWATHI JN for guiding me through this project.

EXECUTIVE SUMMARY

This project implements the super-resolution algorithm proposed by [1] Chih-Yuan Yang and Ming-Hsuan Yang's "Fast Direct Super-Resolution by Simple Functions" Paper published by the University of California at Merced.

- A Python implementation is presented
- A small set of images are selected as training set different from those referred to by the original paper
- Algorithm parameters are tuned down to compensate for the reduced data and images are limited to grayscale for simplicity
- Super-resolution is performed and results obtained for metrics - Structural similarity index and Mean Square Error.
- Untrained images are also tested and results Analysed
- Conclusion and References are provided

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

SISR	Single Image Super Resolution
HR	High Resolution
LR	Low Resolution

NOTATIONS AND SYMBOLS

C	Coefficient matrix
Il	Low Resolution Image
Ih	High Resolution Image
⊗	Convolution Operator
G	Gaussian Kernel
↓	Downsampling Operator
s	Scaling Factor
Ph	High Resolution Patch
Pl	Low Resolution Patch
vi	LR Patch row vector
wi	HR Patch row vector
m	Number of rows in W
n	Number of rows in V
W	HR Patches matrix
V	LR Patches matrix

1. INTRODUCTION

1.1 OBJECTIVE

Image enhancement is the process of adjusting digital images so that the results are more suitable for display or analysis depending on the domain. Image upscaling also known as super-resolution is a form of image enhancement specific to the domain of infotainment. The aim of single image super-resolution is to reconstruct a high-resolution image from a single low-resolution input.

Fig 1.1.1 Image of a flower with different native resolutions scaled to same dimensions

1.2 MOTIVATION

SISR aims to generate a visually pleasing high-resolution HR image from a given low-resolution LR input. It is a challenging and ill-posed problem because numerous pixel intensities need to be predicted from limited input data. Different algorithms have been developed each focusing on different definitions of ‘pleasing’ from smoothing edges to enhancing colors.

With growing native resolutions and hardware improvements, digital content will require upscaling if it is to meet up with the growing generations standards. This is the motivation for my project, to understand and implement a modern image upscaling algorithm.

Fig 1.2.1 Increase in native display sizes of the years

1.3 BACKGROUND

Enhancement vs Upscaling

Image enhancement is the process of adjusting digital images so that the results are more suitable for display or analysis depending on the domain.

Image upscaling also known as super-resolution is a form of image enhancement specific to the domain of infotainment.

The aim of single image super-resolution is to reconstruct a high-resolution image from a single low-resolution input. In a way, it is the inverse of the downsampling operation or pooling.

Patch

Image patch is a container of pixels in larger form. Also known as blocks, they can be of different sizes, usually rectangular with fixed dimensions. They can be overlapping or non overlapping.

In this paper we will be dealing with non overlapping square patches of fixed dimensions. As an example, a 100x100 image with 10x10 patch size will produce 100 patches. It is analogous to a matrix and its submatrices.

Filtering

Filtering is a process of changing each pixel of an image by applying similar operations to it. Usually done through a kernel or window that slides through each pixel and its neighboring pixels and a mathematical operation is performed producing new pixel values. In this paper we will apply a gaussian filter to reduce or downscale images in our training set.

2. PROJECT DESCRIPTION AND GOALS

The main objectives of this project can be broken down into 3 steps.

1. Data Collection and Separation: Dataset will be downloaded and segregated into downscaled and original images.
2. Implementation: The algorithm proposed will be implemented and tested using Python 3.9 and other libraries.
3. Testing and Conclusion: Upscaled images will be compared to original high resolution images under similarity metrics, results tabulated and conclusions drawn.

3. TECHNICAL SPECIFICATION

Language	Python 3.9
Libraries	Scikit-image, Pandas, matplotlib, numpy
Environment	Google Colab
Ram	12GB
Disk	68GB
CPU	1 x single core hyper threaded Xeon Processors @2.3Ghz i.e(1 core, 2 threads)

Table 3.1 Technical specifications

4. DESIGN APPROACH AND DETAILS

The proposed algorithm is a divide and conquer strategy that divides low resolution images into patches and applies a mapping function to the patches to produce high resolution images. In formal terms the objective is to find Coefficient matrix C for each patch.

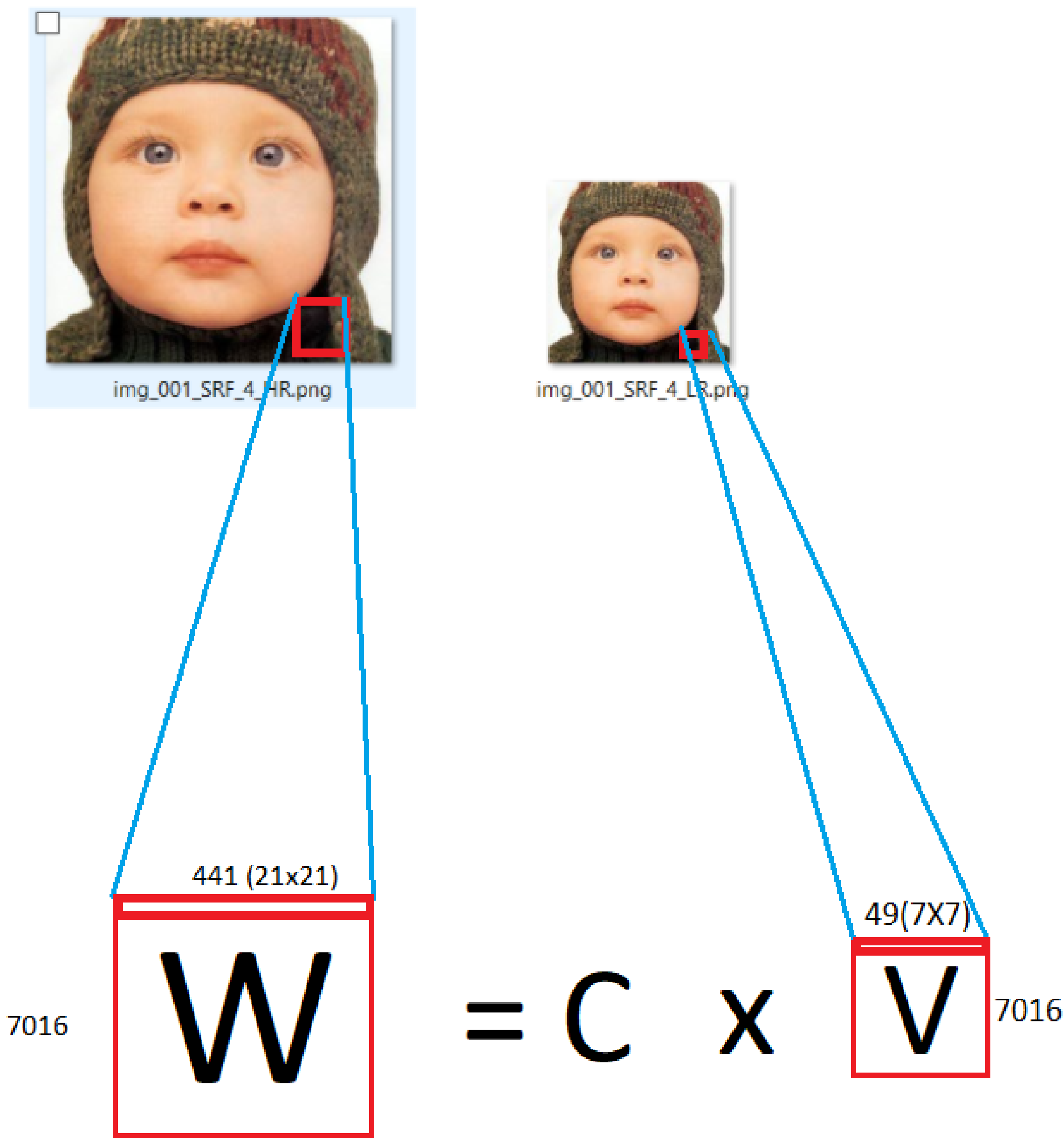

Fig 4.1 Mapping each LR patch to HR patch

1. From dataset and HR Images Ih, LR image Il is generated by applying formula,

   $$Il = (Ih \otimes G) \downarrow s$$

   where $\otimes$ is a convolution operator, G is a Gaussian kernel,
   $\downarrow$ is a downsampling operator and s is the scaling factor.

2. From each Ih and the corresponding Il image, a large set of corresponding HR and LR patch pairs are cropped.

3. Let Ph and Pl be two paired patches. They extract the features of Ph and Pl as the intensities minus μ, the patch mean, to present the high-frequency signals.

4. For HR patch Ph, only extract features for pixels at the central region and discard boundary pixels as the LR patch Pl does not carry sufficient information to predict those pixels.

5. Then they collect a large set of LR patches from natural images to learn K cluster centers of their extracted features.

6. Suppose there are l LR exemplar patches belonging to the same cluster. Let vi and wi (i = 1,...,l) be vectorized features of the LR and HR patches respectively, in dimensions m and n.

7. Final step is to learn a set of n linear regression functions to individually predict the n feature values in HR.
   - Let V ∈ Rm×l and W ∈ Rn×l be the matrices of vi and wi
   - Regression coefficients matrix C is calculated by solving the least squares problem:

     $$\mathbf{C}^* = \underset{\mathbf{C}}{\operatorname{argmin}} \left\| \mathbf{W} - \mathbf{C} \begin{pmatrix} \mathbf{V} \\ \mathbf{1} \end{pmatrix} \right\|^2$$

   - The HR patch W is then calculated by multiplying C with V

     $$\mathbf{w} = \mathbf{C}^* \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix}$$

5. PROJECT DEMONSTRATION

SR Set 5[6] and Set 14[7] Datasets will be used as training image data, thus a total of 13 images which gives 7016 patches.

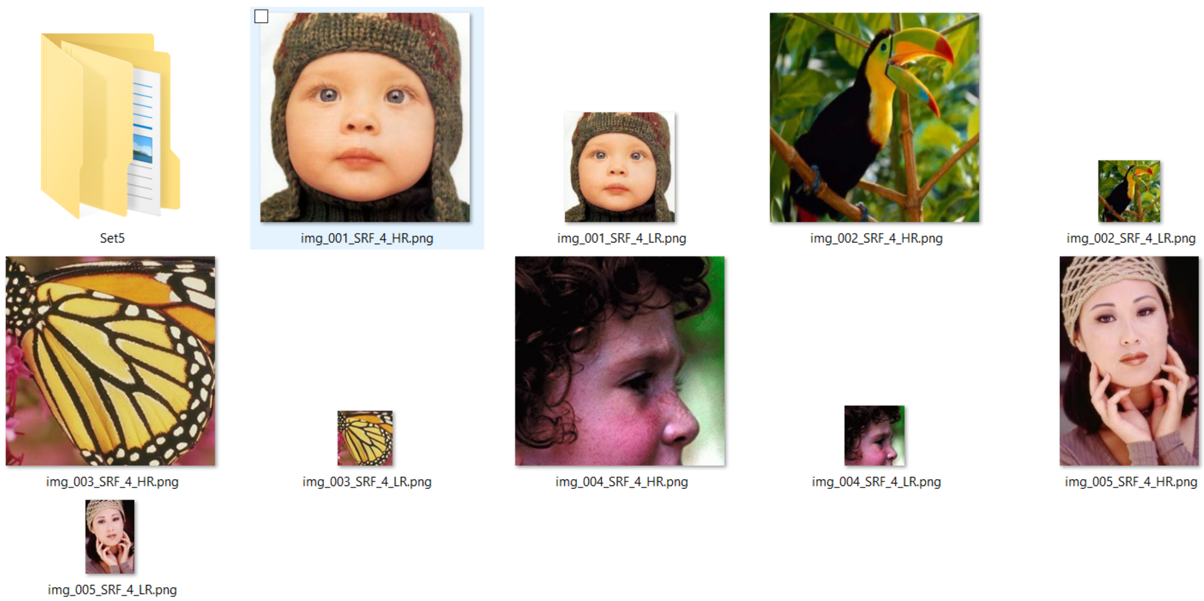

Fig 5.1 Image HR LR pairs in dataset

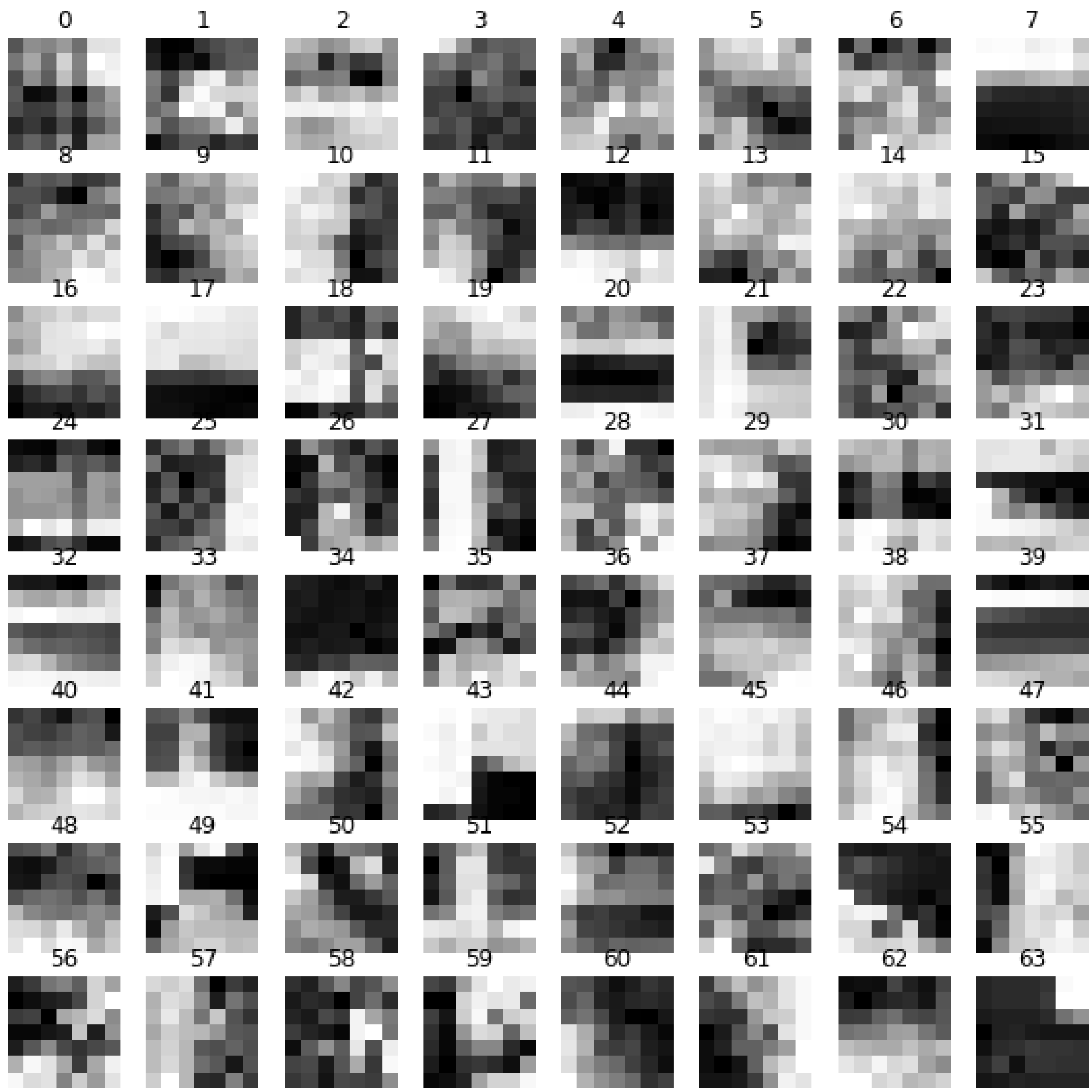

Fig 5.2 64 Cluster centers trained using K Means

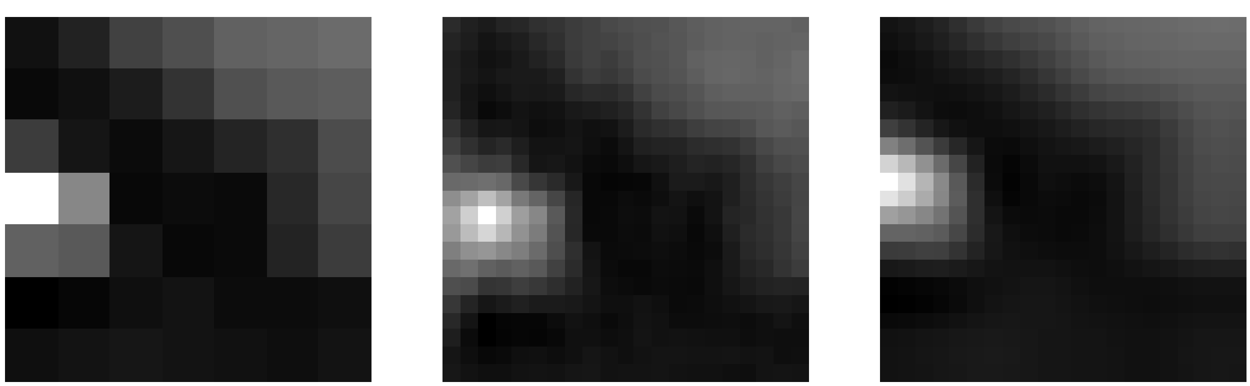

Fig 5.3 LR patch in left, Proposed algorithm middle, Interpolation right

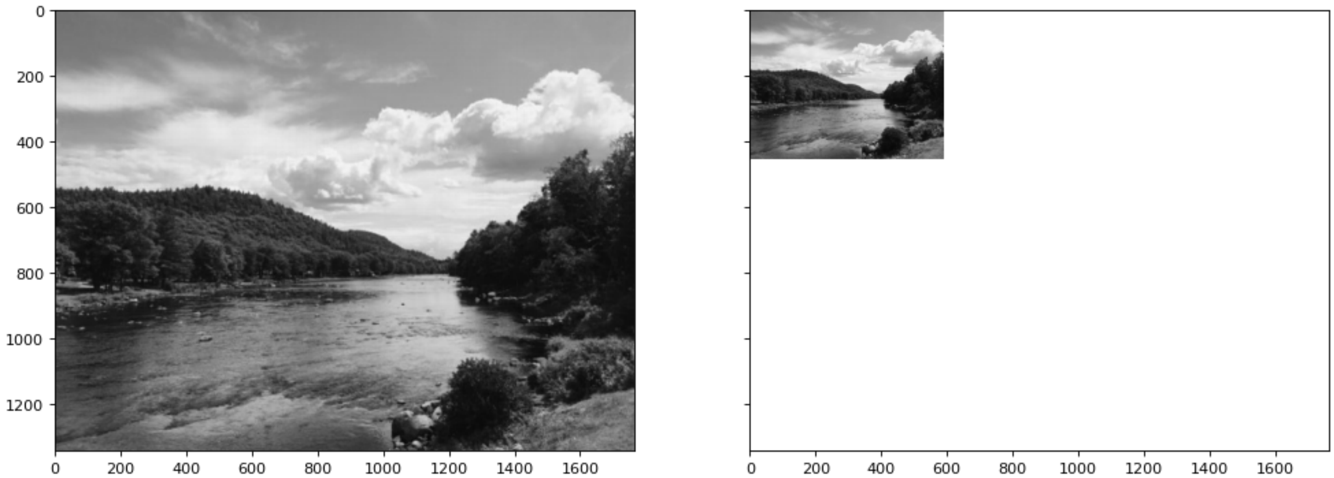

Fig 5.4 Untrained image on right, Upscaled result left

6. RESULTS AND CONCLUSION

PERFORMANCE VS TRAINING IMAGE SET

IMAGE	SSIM	MSE
1	0.616606	0.009822
2	0.806730	0.004400
4	0.713286	0.004578
5	0.747840	0.008731
6	0.867087	0.000999
7	0.833176	0.004018
8	0.917116	0.002325
9	0.891794	0.001503
10	0.799559	0.003855
11	0.928247	0.002361
12	0.900828	0.002178
13	0.875278	0.007784
14	0.833600	0.004454

Table 6.1 Performance metrics against training image set

The table shows the Structural Similarity Index and Mean Squared Error for all images in the training image set as predicted by the algorithm. An average of 0.825 SSIM is not bad though it can be better. MSE is on average 0.0043 which is a very good score.

The algorithm does what it was meant to do, it upscales without disturbing or changing the images very much and does obtain a very close prediction to the high resolution image.

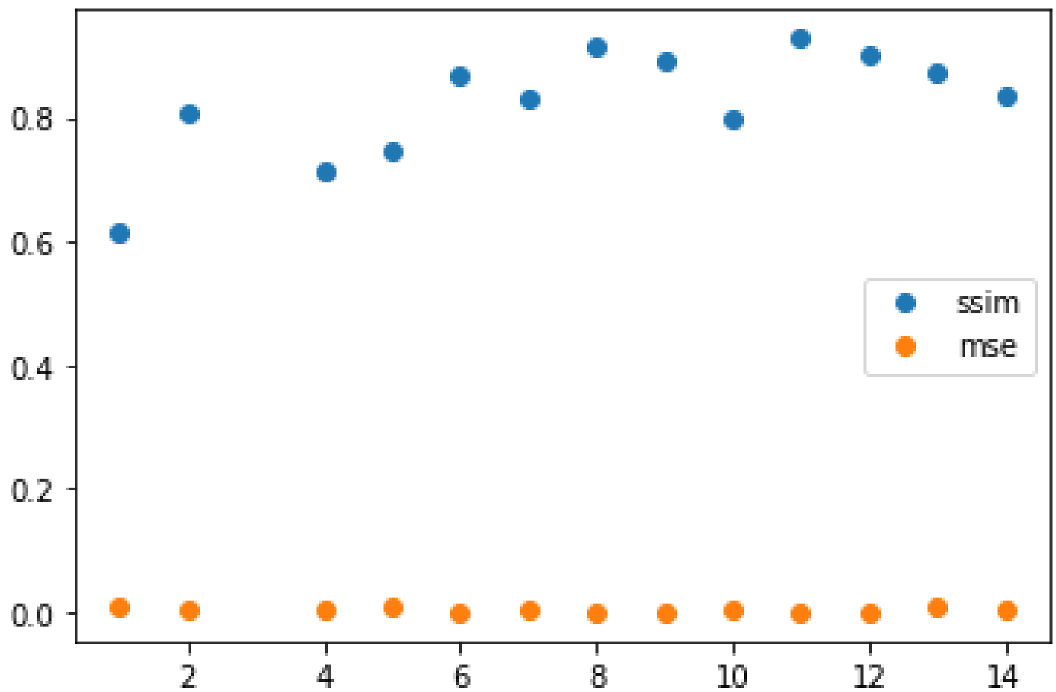

Figure 6.1 Performance graph

However, the original paper presented much better results. Due to data size constraints and greyscale restriction, those values could not be obtained. But with necessary resources the same experiment can be repeated with a large training dataset on color images and huge untested data.

REFERENCES

[1] C.-Y. Yang and M.-H. Yang. Fast Direct Super-Resolution by Simple Functions. In ICCV, 2013

[2] Giachetti, Andrea, and Nicola Asuni. "Real-time artifact-free image upscaling." IEEE Transactions on Image Processing 20.10 (2011): 2760-2768.

[3] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In ICCV, 2009

[4] C. Dong, C. Change Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In ECCV, 2014

[5] Schulter, Samuel, Christian Leistner, and Horst Bischof. "Fast and accurate image upscaling with super-resolution forests." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[6] https://uofi.box.com/shared/static/kfahv87nfe8ax910l85dksyl2q212voc.zip

[7] https://uofi.box.com/shared/static/igsnfieh4lz68l926l8xbklwsnnk8we9.zip

[8] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

[9] http://labelme.csail.mit.edu/Release3.0/browserTools/php/dataset.php