Image Inpainting using Regression

(B.Tech Thesis)

Ashish Kumar , Smriti Jain

Mentor- Dr. Gaurav Bhatnagar

Presented at Indian Institute of Technology, Jodhpur
on 6 May 2015

Motivation

- Inpainting is the process of reconstructing lost regions in an image or removing unwanted objects in an image .

Applications:

- repairing cracks in images, and object or text removal.
- for fast image compression

Contributions

- We developed a framework for a 2 phase inpainting algorithm which give the following results for a dataset of 14 gray scale images:
- PSNR values are above 59 .
- UIQ values is close to 1.
- SSIM index is close to 1.
- Our Algorithm works on gray scale images of any size.
- The region to be inpainted should contain white pixel values

Problem Description

- Given a gray scale image I of size MxN, and an unknown region $\Omega \subset I$
- We have to determine:

$$Intensity\ at\ (i, j)\ \forall\ \mathrm{all}\ (i, j) \in \Omega$$

with the help of known pixels of the image.

Reducing Inpainting into Regression problem

- We formulated the inpainting problem as a 3-D regression problem.
- The proposed regression framework relates Intensity I to a function of x and y.

$$I \approx f(x, y)$$

,where x and y represent x and y coordinate of a pixel value.

- The objective is to approximate the known training dataset with a function, f(x,y), and then use it to predict unknown pixels to determine its value.

Why Regression ?

- Certain known pixels are a part of a thin line or other similar structures, which may not affect its certain neighbouring pixels.
- Consequently, these pixels become noisy data.

- Thus, when compared to interpolation, the regression framework is more appropriate for our setup.

Proposed Technique

- The proposed algorithm uses block-wise regression to fill the unknown pixel values in the input image F. The block sizes are variable and depend on the extended edge map of the image.

- The entire algorithm is divided into two modules ---

(I). Dividing Image into sub-blocks

- The entire image F is divided into similar regions or rectangular sub-blocks .

- We determine these sub-blocks from the linearly extended edge-map of the input image with the property that no sub-image contains an edge.

- The desired level of similarity of a sub-block is controlled by tuning the gaussian parameter of canny edge detector.

(II). Applying Regression on each sub-block

- The inpainting problem in each sub-block is solved by learning a Regression function on the known data of the image.

- We show the result for the following two regression techniques:

K-NN regression

Support Vector Regression

K-nn Regression

- K-nearest neighbours Regression predicts the target values of missing pixels based on a similarity measure. Distance functions are used as similarity measure in our case.
- Following Euclidean distance function is minimized to predict the unknown values:

$$\sum_{i=1}^{k}(X_i - I_i)^2$$

,where k is the number of neighbours

X_i represents both the coordinates x and y of neighbour i and

$\boldsymbol{I_i}$ represents Intensity at X_i

Support Vector Regression

- The benefit of using SVM for regression is that they exploit sparsity for generating predictions, which improves the computational complexity in the case of handling of large images.
- Lets say X represents both the dependent variables x and y. Thus, following regularized ε-sensitive error function is minimized to approximate f(X).

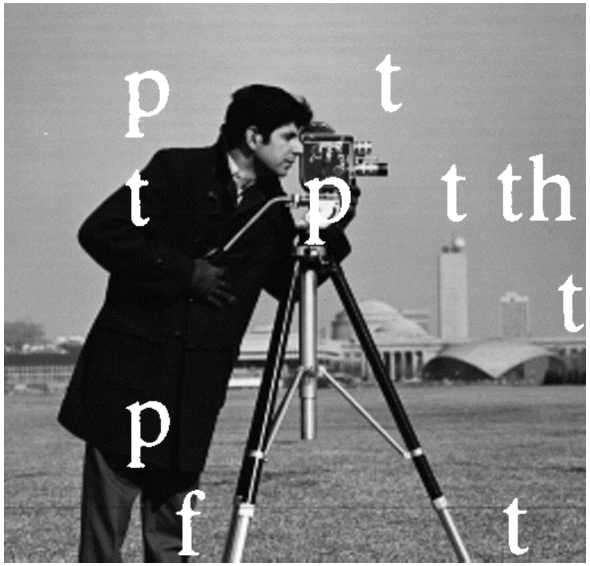
$$C = \sum_{n=1}^{N} E_{\in}(f(X_n) - I_n) + \frac{1}{2}\|w\|^2$$

- N total number of known pixels in a sub block
- X_n denotes coordinate (x_n, y_n) of nth pixel
- I_n denotes the target value that is the intensity value of pixel at location

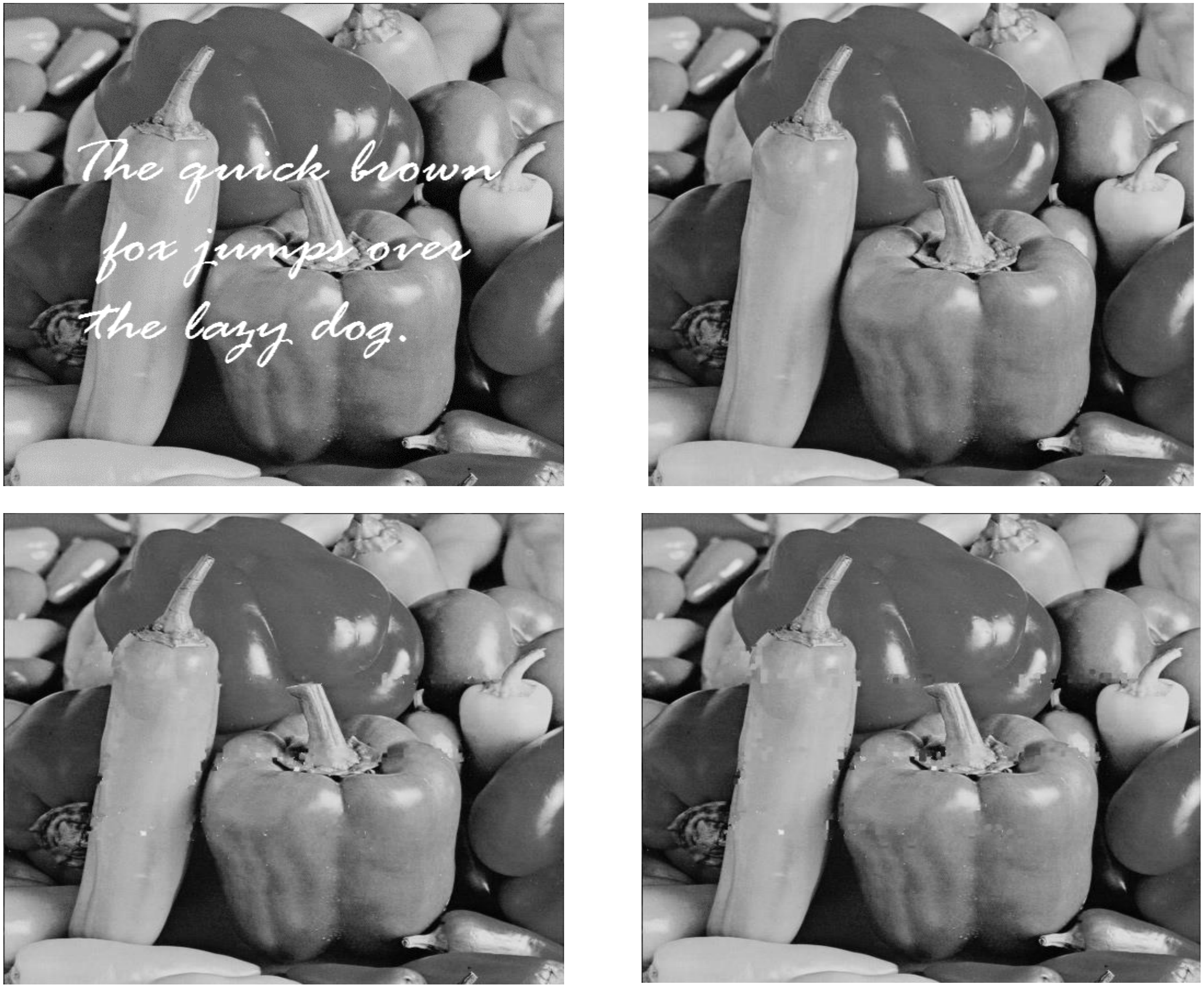
Experiments and results

- We use python 2.7 for implementing our algorithm and test it on a dataset of 14 gray scale images. We evaluate the result images on the following metrics:
- **SSIM**: Spatial Frequency (SF) measures the overall activity level in an image .If the value of SF becomes higher after inpainting then the activity level will increase.
- **PSNR**:The PSNR indicates the similarity between two images.The higher the value of PSNR, the better the inpainted image is.
- **UIQ:**The UIQ indicates the structural similarity between two images.

Textual and Random Loss



Textual and Random Loss



Textual and Random Loss

Textual and Random Loss




Images with natural Data Loss

Metric Values

Images	PSNR		UIQ		SSIM	
	SVR	K-NN	SVR	K NN	SVR	K NN
monty.jpg	61.13	49.96	0.99	0.99	0.99	0.99
lake.jpg	59.76	49.76	0.99	0.99	0.99	0.99
elaine.jpg	61.39	49.96	0.99	0.99	0.99	0.99
cameraman.jpg	60.46	50.51	0.99	0.99	0.99	0.98
elaine2.jpg	62.41	49.96	0.99	0.99	0.99	0.99
Livingroom.jpg	66.73	49.95	0.99	0.998	0.99	0.99
monalisa.jpg	54.21	50.57	0.83	0.83	0.83	0.83
baboon.jpg	59.16	50.28	0.98	0.98	0.98	0.98
EINSTEIN.jpg	55.32	49.43	0.92	0.92	0.92	0.92
7101.gif	63.85	49.94	0.98	0.99	0.92	0.99
peppers.gif	61.46	49.92	0.99	0.98	0.98	0.98
peppersgray.gif	61.61	50.95	0.99	0.99	0.99	0.99
lena_iitj.png	60.14	57.41	0.99	0.98	0.99	0.99

Conclusions and Future Work

In Conclusion, we see that the 2 phase inpainting framework gives good results for linear edge extension with SVR or k-nn

For future work, the linear edge extension is too simplistic and tends to give erroneous results for certain images. It could be replaced by better edge extension techniques. Also, the restriction of having rectangular sub-blocks can be relaxed.

Thank You