WEB IMAGE INTERPOLATION VIA WEIGHTED TOTAL LEAST SQUARES REGRESSION

Xianming Liu[1], *Deming Zhai*[1], *Guangtao Zhai*[2], *Debin Zhao*[1], *Wen Gao*[2]

[1]School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
[2]School of Electrical Engineering and Computer Science, Peking University, Beijing, 100871, China

ABSTRACT

Although ordinary least squares (OLS) regression achieves great success in clean image interpolation, its effectiveness is questionable in the scenario of web images which are usually compressed beforehand. The inherent flaw of OLS is that it is asymmetric, the perturbation is only confined on the right side of the linear system. It is not reasonable for web images. Considering the drawback of OLS, in this paper, we propose an efficient web image interpolation algorithm based on total least squares (TLS) regression. In the proposed method, small perturbations are allowed in both side of the system, which are optimized by TLS in a patch-based manner. Furthermore, we develop a weighted version of TLS to consider contribution diversity of different samples and patches in model estimation, which can efficiently remove the influence of outliers in regression. Experimental results on benchmark test images demonstrate the efficiency of our method.

***Index Terms*—** Web image interpolation, ordinary least squares, total least squares

1. INTRODUCTION

With the rapid development of social network and internet technology, more and more people would like to share their photos in websites and retrieve their wanted images by search engines. Due to the limitation of bandwidth and server storage, in many cases, web images need to be downsampled and compressed, and only when concerned they are then upsampled to the original resolution to show more information to the user. Image retrieval is a typical example. When we input a keyword, the search engine returns a lot of related images, which are shown in their low-resolution (LR) form in order to give users more choice to choose their really interested one. When users click the chosen image, the original high-resolution (HR) image will be transmitted from the server. If the original one is not available, image interpolation technology is needed to recover the LR image to its original resolution. This case often happens due to the limitation of server storage space or congestion of internet. From the above example, we can find image interpolation technology plays an important role in web image services.

Image interpolation itself is a very active research topic in image processing. In the literature, many image interpolation algorithms have been proposed [1-6]. Bilinear and bicubic are two popular global methods based on classical data-invariant linear filters. In [1], Zhang and Wu propose to interpolate a missing pixel in multiple directions, and then fuse the directional results by minimum mean square-error estimation.

Due to the power of image modeling, linear auto-regression (AR) model based methods have received more and more attention, which integrate edge direction information into AR model parameters. There are two representative works in the literature: NEDI and SAI. They are both based on the geometric duality between the LR and HR covariance. The NEDI method, proposed by Li and Orchard [2], estimates local covariance coefficients from a LR image and then project the estimated covariance to the HR image to adapt the interpolation procedure. Zhang and Wu propose the SAI algorithm [3], which learns and adapts varying scene structures using a locally linear regression model, and interpolates the missing pixels in a group by a soft-decision manner. NEDI and SAI achieve promising results in both objective and subjective performance. However, both NEDI and SAI only consider quality degradation by downsampling, and never consider the influence of compression noise on estimation of AR model parameters. This is not reasonable for practical applications of web images interpolation, where compression is widely adopted.

In the literature, Xiong *et al.* [4] also consider the problem of compressed image interpolation, and propose a robust web image/video super-resolution algorithm in a learning-based manner. However, it needs an additional training image set for offline learning. Zhang *et al.* in [5] propose a joint denoising and zooming algorithm under the LMMSE framework, which is based on the assumption that the noise is additive white Gaussian noise. However, for compressed noise, this assumption is usually not satisfied.

Essentially, NEDI and SAI are both based on ordinary least squares (OLS) regression, which is in a asymmetry manner and not suitable for compressed image interpolation. Considering the drawback of OLS, we propose an efficient algorithm based on total least squares (TLS) regression in an online manner. In the proposed method, small perturbations are allowed in both side of the system. Furthermore, we develop a weighted version of TLS to consider contribution diversity of different samples and patches in model estimation, which can efficiently remove the influence of outliers in regression.

The rest of this paper is organized as follows. In Section 2, we give a brief description of OLS and TLS. Section 3 discusses the algorithm details. Experimental results are presented in Section 4. Section 5 concludes the paper.

2. PROBLEM DESCRIPTION

Image interpolation can be regarded as a data fitting problem. Conceptually and algorithmically, fitting linear models to data can be achieved by solving a system of equations:

$$\mathbf{A}\mathbf{w} = \mathbf{b}, \tag{1}$$

where the matrices $\mathbf{A}$ is the data matrix and the vector $\mathbf{b}$ is the observation vector. They are constructed from the given data. The vector $\mathbf{w}$ represents the parameters of the to-be-found model. NEDI and SAI exploit the OLS manner to address such problem. Despite great diversity in implementation, they invariably consider solving approximately an overdetermined system of equations.

The ordinary least squares approximation is obtained as a solution of the following optimization problem:

$$\begin{aligned} \{\hat{\mathbf{w}}_{LS}, \Delta\mathbf{b}_{LS}\} &= \arg\min_{\mathbf{w},\Delta\mathbf{b}} \|\Delta\mathbf{b}\|_2 \\ &\text{subject to } \mathbf{A}\mathbf{w} = (\mathbf{b} + \Delta\mathbf{b}). \end{aligned} \tag{2}$$

The rationale behind this approximation method is to correct the right-hand side $\mathbf{b}$ as little as possible in the L_2 norm sense. And we can obtain an optimal solution with a close form as:

$$\hat{\mathbf{w}}_{LS} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{b}. \tag{3}$$

Although ordinary least squares regression achieves great success in clean image interpolation, its effectiveness is questionable in the scenario of compressed images. The inherent flaw of ordinary least squares is that it is asymmetry, the perturbation is only confined to the right side $\mathbf{b}$ and consider the left side $\mathbf{A}$ is noise-free. It is not true for compressed image, since there also is compression noise in the data matrix $\mathbf{A}$.

Instead, in this paper, we propose to develop an efficient compressed image interpolation algorithm based on total least squares (TLS), in which small perturbations are allowed in both side of the system. The total least squares (TLS)[7][8] problem can be written as:

$$\begin{aligned} \{\hat{\mathbf{w}}_{TLS}, \Delta\mathbf{A}, \Delta\mathbf{b}\} &= \arg\min_{\mathbf{w},\Delta\mathbf{A},\Delta\mathbf{b}} \|[\Delta\mathbf{A}\ \Delta\mathbf{b}]\|_F \\ &\text{subject to } (\mathbf{A} + \Delta\mathbf{A})\mathbf{w} = (\mathbf{b} + \Delta\mathbf{b}) \end{aligned} \tag{4}$$

where the subscript F denotes the Frobenius norm, and $\Delta\mathbf{A}$ and $\Delta\mathbf{b}$ are the perturbations on the left and right side of the linear system respectively. Supposing the singular value decomposition of the data matrix $[\mathbf{A}\ \mathbf{b}]$ is

$$\begin{aligned} [\mathbf{A}\ \mathbf{b}] &= U\Sigma V^T \qquad (5) \\ &= [U_{\mathbf{A}}\ U_{\mathbf{b}}] \begin{bmatrix} \Sigma_{\mathbf{A}} & 0 \\ 0 & \Sigma_{\mathbf{b}} \end{bmatrix} \begin{bmatrix} V_{\mathbf{AA}} & V_{\mathbf{Ab}} \\ V_{\mathbf{bA}} & V_{\mathbf{bb}} \end{bmatrix}^T \end{aligned}$$

where the matrices are partitioned according to the dimensions of $\mathbf{A}$ and $\mathbf{b}$. The close form solution to the general TLS problem in (4) is

$$\hat{\mathbf{w}}_{TLS} = -V_{\mathbf{Ab}}V_{\mathbf{bb}}^{-1}. \tag{6}$$

3. WEIGHTED TOTAL LEAST SQUARES REGRESSION FOR INTERPOLATION

In this section, we develop a practical compressed image interpolation algorithm based on weighted total least squares regression. First, we will introduce the interpolation model used in our method, then multi-scaled image patch selection strategy is presented, at last the weighted total least squares regression is detailed.

3.1. Interpolation Model

Suppose we are given a compressed LR image L, we want to get its corresponding HR image H. We first give a initial estimation $\hat{H}$ of the original HR image by using some simple image interpolation algorithms, such as bicubic. Let x be the current unknown pixel to be estimated. We crop an image patch centered on x from $\hat{H}$ and re-order it into a column vector denoted by $\mathbf{b} \in R^m$. Let $\mathbf{B} = \{\mathbf{b}_i\}_{i=1}^{j=n} \in R^{m\times n}$ be a collection of image patches from L and $\hat{H}$ in the local neighborhood of the current pixel x. We build the relationship between the LR image and its corresponding original HR image in a linear combination manner:

$$\mathbf{B}\mathbf{w} = \mathbf{b}, \tag{7}$$

where $\mathbf{w} \in R^n$ is the model parameter vector. In practical image interpolation applications, in general, we cannot find such $\mathbf{w}$ making the above equation true. Therefore, we relax the problem and allow small perturbations in both $\mathbf{b}$ and $\mathbf{B}$:

$$(\mathbf{B} + \Delta\mathbf{B})\mathbf{w} = \mathbf{b} + \Delta\mathbf{b}, \tag{8}$$

where $\Delta\mathbf{b} \in R^m$ and $\Delta\mathbf{B} \in R^{m\times n}$ are the perturbations in $\mathbf{b}$ and $\mathbf{B}$, respectively. It is reasonable for compressed image interpolation since both the data matrix $\mathbf{B}$ and the observation vector $\mathbf{b}$ have compressed noise.

Furthermore, $\mathbf{B}$ and $\mathbf{b}$ can be centralized as

$$\widetilde{\mathbf{b}}_i = \mathbf{b}_i - \widehat{\mathbf{b}}_i, \widetilde{\mathbf{b}} = \mathbf{b} - \widehat{\mathbf{b}}, \tag{9}$$

where $\widehat{\mathbf{b}}_i$ and $\widehat{\mathbf{b}}$ are average vector with each elements are the average value of $\mathbf{b}_i$ and $\mathbf{b}$, respectively. Therefore, we can solve for $\mathbf{w}$ by addressing the following optimization problem:

$$\begin{aligned} \{\hat{\mathbf{w}}_{TLS}, \Delta\mathbf{B}, \Delta\mathbf{b}\} &= \arg\min_{\mathbf{w},\Delta\mathbf{B},\Delta\mathbf{b}} \|[\Delta\mathbf{B}\ \Delta\mathbf{b}]\|_F. \\ &\text{subject to } (\widetilde{\mathbf{B}} + \Delta\mathbf{B})\mathbf{w} = (\widetilde{\mathbf{b}} + \Delta\mathbf{b}) \end{aligned} \tag{10}$$

3.2. Multi-scaled Image Patch Selection

In the interpolation model described above, the observation vector $\mathbf{b}$ is regarded as a linear combination of basis vectors, $\{\mathbf{b}_i\}$. Therefore, $\{\mathbf{b}_i\}$ plays a very important role in model estimation, and they should be reasonably similar to $\mathbf{b}$ in order to capture image features in $\mathbf{b}$. We exploit a multi-scaled manner to collect image patches based on the geometry duality between the LR image and the HR one. Therefore, intra-scale and inter-scale correlation are exploited simultaneously to achieve accurate model estimation.

First, we consider the correlation across different scales, and we collect image patches $\{\mathbf{b}_i^L\}$ from the LR image in a local neighborhood around x. Second, we take into account the correlation in the same scale. Then, image patches $\{\mathbf{b}_i^H\}$ are cropped from the estimated HR image $\hat{H}$ in a local neighborhood centered in x. In this way, a set of image patches $\{\mathbf{b}_i\}$ including $\{\mathbf{b}_i^L\}$ and $\{\mathbf{b}_i^H\}$ is constructed. Note that all of chosen patches are with the same size as $\mathbf{b}$. With such a procedure, the local statistics of the variables can be accurately computed so that the image edge structures can be well preserved.

3.3. Weighted Total Least Squares

In the optimization procedure defined in subsection 3.1, different patches are considered equally important to determine the model parameters. However, the similarity between different patches to the current patch are diversity. We should incorporate such difference in minimization to achieve more accurate model estimation.

We introduce $\mathbf{W}_{intra}$ as the intra correlation matrix to account for the similarity between each pixel in one patch and the center pixel which is the one being estimated. In particular, the Radial Basis Function (RBF) kernel is utilized to compute intra weights decreasing with distance from the neighborhood center, as formulated as follows.

$$\phi_l = \exp\left\{-||\mathbf{x}_l - \mathbf{x}_0||^2/\sigma_s^2\right\}, \sigma_s > 0, 0 \le l \le m-1 \quad (11)$$

where $\mathbf{x}_0$ and $\mathbf{x}_l$ are location vectors of the center pixel and the l^{th} pixel in one patch, and σ_s is a bandwidth parameter with the value 0.5. As such the diagonal weight matrix $\mathbf{W}_{intra}$ reflects the quality of the data in the local neighborhood through measuring its ℓ_2 distance to the central model. The smaller is the distance the bigger is the weight assigned to the row of the data. Another matrix $\mathbf{W}_{inter}$ is also introduced to reflect the inter correlation between different patches with the observation vector $\mathbf{b}$. Similarly, we can define the column weight matrix $\mathbf{W}_{inter} = \mathrm{diag}(\psi_1, \psi_2, \ldots, \psi_n)$, with

$$\psi_i = \exp\left\{-G \cdot ||\mathbf{b}_i - \mathbf{b}||^2/\sigma_p^2\right\}, \sigma_p > 0, 1 \le i \le n \quad (12)$$

where G is a Gaussian kernel used to take into account the distance between the central pixel and other pixels in the patch, and $\mathbf{b}_i$ represents the pixel vector re-ordered from the i^{th} patch, and σ_p is a bandwidth parameter with the value 0.5. This patch comparison permits a reliable similarity measure involving pixels which can fall far away from each other. And this weighting process help to penalize larger models so as to prevent model overfitting.

With the intra and inter correlation matrix, we can replace the Frobenius norm in (4) into a weighted matrix norm to express the relative importance of different pixel and patch in estimating the parameters of the model.

$$\{\hat{\mathbf{w}}_{WTLS}, \Delta\mathbf{B}, \Delta\mathbf{b}\} = \underset{\mathbf{w}, \Delta\mathbf{B}, \Delta\mathbf{b}}{\arg\min} \|\mathbf{W}_{intra}[\Delta\mathbf{B}\ \Delta\mathbf{b}]\mathbf{W}_{inter}\|_F \quad (13)$$

$$\text{subject to } (\widetilde{\mathbf{B}} + \Delta\mathbf{B})\mathbf{w} = (\widetilde{\mathbf{b}} + \Delta\mathbf{b})$$

where $\mathbf{W}_{intra}$ weights the rows and $\mathbf{W}_{inter}$ weights the columns of $[\Delta\mathbf{B}\ \Delta\mathbf{b}]$. Therefore we can choose the row and column weight to reflect the properties of the data in the matrix $[\Delta\mathbf{B}\ \Delta\mathbf{b}]$.

Based on the TLS solution given in (6), the WTLS problem in (13) has a close form solution as

$$\hat{\mathbf{w}}_{WTLS} = \left(\sqrt{\mathbf{W}_{intra}}\right)^{-1}(-V_{\mathbf{Ab}}V_{\mathbf{bb}}^{-1})\left(\sqrt{\mathbf{W}_{inter}}\right)^{-1}. \quad (14)$$

With the WTLS solution, we can get the prediction $\bar{\mathbf{b}}$ and pick up the center value as the estimation of the current pixel x.

4. EXPERIMENTAL RESULTS

In this section, experimental results are presented to verify the performance of the proposed algorithm. For comparison, we test three other interpolation methods, including DFDF [2], NEDI [3], and SAI [4]. We select several standard test images of size 512×512, as depicted in Fig. 1.

In our experiments, we first downsample each image by a factor of two to get the LR image, which is further compressed by JPEG with three quality factors (QF): 60, 70 and 80. Note that QF is chosen from $[0, 100]$, and 100 represents the highest quality image without any compression. Then, the results are interpolated back to their original sizes.

Fig. 1. Five sample images in the test set.

Let us consider the objective and subjective quality comparison of five algorithms. We quantify the average objective quality over the whole image by PSNR, and we exploit EPSNR to focus on fidelity of image edges. In our study, the Sobel edge filter is used to locate the edge in the original image, and the PSNR of the pixels on the edge are used to generate the EPSNR [6]. Although PSNR and EPSNR can measure

Table 1. Objective and subjective quality comparison of four interpolation algorithms for compressed images

Method /QF	DFDF			NEDI			SAI			TLS		
	PSNR	EPSNR	SSIM	PSNR	EPSNR	SSIM	PSNR	EPSNR	SSIM	PSNR	EPSNR	SSIM
Airplane /60	28.96	19.91	0.8828	27.79	18.37	0.8769	28.74	19.85	0.8802	29.07	20.02	0.8837
/70	29.25	19.97	0.8944	28.06	18.47	0.8884	29.31	19.93	0.8945	29.41	20.14	0.8953
/80	29.41	20.02	0.9004	28.16	18.51	0.8941	29.49	19.98	0.9015	29.56	20.19	0.9012
average	29.20	19.96	0.8925	28.01	18.45	0.8864	29.18	19.92	0.8921	**29.35**	**20.11**	**0.8934**
Baboon /60	21.79	18.12	0.6364	21.75	17.74	0.6393	21.95	18.26	0.6482	22.18	18.41	0.6527
/70	21.96	18.23	0.6561	21.97	17.84	0.6613	22.18	18.42	0.6709	22.42	18.57	0.6757
/80	22.05	18.31	0.6657	22.09	17.96	0.6718	22.31	18.51	0.6814	22.53	18.67	0.6863
average	21.93	18.22	0.65273	21.93	17.84	0.6574	22.14	18.39	0.6668	**22.38**	**18.55**	**0.6716**
Couple /60	27.92	23.46	0.7859	27.57	22.94	0.7761	27.94	23.33	0.7867	28.05	23.61	0.7888
/70	28.22	23.58	0.8009	27.85	23.19	0.7905	28.31	23.61	0.8035	28.38	23.84	0.8043
/80	28.37	23.67	0.8084	27.96	23.23	0.7977	28.48	23.66	0.8118	28.53	23.93	0.8121
average	28.17	23.57	0.7984	27.79	23.12	0.7881	28.24	23.53	0.8006	**28.32**	**23.79**	**0.8017**
Elaine /60	30.89	28.08	0.7302	30.83	27.41	0.7342	30.82	28.06	0.7263	30.98	28.14	0.7331
/70	30.96	28.29	0.7321	30.91	27.56	0.7365	30.87	28.33	0.7268	31.08	28.37	0.7361
/80	30.97	28.53	0.7325	30.91	27.76	0.7375	30.85	28.55	0.7261	31.11	28.58	0.7373
average	30.94	28.31	0.7316	30.88	27.57	**0.7361**	30.84	28.31	0.7264	**31.06**	**28.36**	0.7355
Peppers /60	30.67	22.47	0.8281	28.76	18.32	0.8252	30.53	21.93	0.8265	30.73	22.51	0.8282
/70	31.01	22.61	0.8371	28.97	18.34	0.8347	30.89	22.11	0.8364	31.08	22.67	0.8374
/80	31.18	22.67	0.8421	29.09	18.41	0.8393	31.06	22.16	0.8414	31.27	22.77	0.8424
average	30.95	22.58	0.8357	28.94	18.35	0.8330	30.82	22.06	0.8347	**31.03**	**22.65**	**0.8361**

the intensity difference between two images, it is well-known that they may fail to describe the visual perception quality of the image. We further choose SSIM as the metric for image visual quality assessment.

Table 1 tabulates the objective and subjective quality comparison. It is clearly seen that for all instances the proposed algorithm consistently works better than other three methods in terms of average PSNR. For all cases, our method also achieves best performance measured by EPSNR, which means the proposed method can better preserve edge structure. We also give the subjective quality comparison with respect to SSIM. It can be easily observed the proposed algorithm produces better performance compared with SAI. Such results clearly demonstrate the superiority of the proposed method in reconstructing the high frequency, such as edges and textures, from compressed LR images.

5. CONCLUSION

In this paper, we present an efficient compressed image interpolation algorithm based on total least squares (TLS) regression. Different from ordinary least squares, in our method, small perturbations are allowed in both side of the system, which are optimized by TLS in a patch-based manner. Furthermore, weighted TLS is developed to consider contribution of different samples and patches in model estimation, which can efficiently remove the influence of outliers in regression. Experimental results demonstrate that our method achieves very competitive performance compared the state-of-the-art methods in both objective and subjective quality.

6. ACKNOWLEDGEMENT

This work was supported in part by the National Science Foundations of China (60736043, 61073083, 61001145), in part by the Major State Basic Research Development Program of China (973 Program 2009CB320905).

7. REFERENCES

[1] L. Zhang and X. Wu, “Image interpolation via directional filtering and data fusion,” IEEE Trans. Image Process., vol.15, no.8, pp.2226-2238, Aug.2006.

[2] X. Li and M. T. Orchard, “New edge-directed interpolation,” IEEE Trans.Image Process., vol.10, no.10, pp.1521-1527, Oct.2001.

[3] X. Zhang, X. Wu, “Image Interpolation by Adaptive 2-D Autoregressive Modeling and Soft-Decision Estimation,” IEEE Trans. Image Process., vol.17, no.6, pp.887-896, June 2008.

[4] Z. Xiong, X. Sun, F. Wu, “Robust Web Image/Video Super-Resolution,” IEEE Trans. Image Process., vol.19, no.8, pp.2017-2028, 2010.

[5] L. Zhang, X. Li and D. Zhang, “Image Denoising and Zooming under the LMMSE Framework,” to appear in IET Image Processing.

[6] X. Liu, D. Zhao, R. Xiong, S. Ma, W. Gao and H. Sun, “ Image Interpolation Via Regularized Local Linear Regression,” IEEE Trans. on Imag. Process., vol.20, no.12, pp.3455-3469, Dec.2011

[7] I. Markovsky, S. Van Huffel, “Overview of total least squares methods,” Signal Processing, pp. 2283-2302, 2007.

[8] K. Hirakawa, T.W. Parks, “ Image denoising using total least squares,” IEEE Trans. Image Process., vol. 15, no.9, pp. 2730-2742, 2006.