IMAGE SUPER-RESOLUTION BY EXTREME LEARNING MACHINE

Le An, Bir Bhanu

Center for Research in Intelligent Systems, University of California, Riverside
lan004@ucr.edu, bhanu@cris.ucr.edu

ABSTRACT

Image super-resolution is the process to generate high-resolution images from low-resolution inputs. In this paper, an efficient image super-resolution approach based on the recent development of extreme learning machine (ELM) is proposed. We aim at reconstructing the high-frequency components containing details and fine structures that are missing from the low-resolution images. In the training step, high-frequency components from the original high-resolution images as the target values and image features from low-resolution images are fed to ELM to learn a model. Given a low-resolution image, the high-frequency components are generated via the learned model and added to the initially interpolated low-resolution image. Experiments show that with simple image features our algorithm performs better in terms of accuracy and efficiency with different magnification factors compared to the *state-of-the-art* methods.

Index Terms— Image, super-resolution, feature, learning

1. INTRODUCTION

Often due to the various limitations such as moderate imaging sensors, the environmental conditions, or the limited transmission channel capacity, the images that we acquire are of low-resolution (LR). The generation of the LR images is commonly modeled with a blurring process followed by downsampling. Visually the LR images are blurred with loss of details in structures that usually reside in the high-frequency (HF) components of the original high-resolution (HR) image. Image super-resolution (SR) consists of the approaches that try to solve the inverse problem of recovering the HR images from the LR images. For some applications where the quality of the images greatly affects the subsequent processing, SR as a preprocessing step is quite desirable. For example, face images in surveillance cameras often have poor resolution, which makes the face recognition algorithms difficult to achieve high accuracy. By using super-resolution (SR) technique to feed HR images to the recognition algorithm, the recognition accuracy can be improved significantly [1].

There has been extensive work on SR methods. Traditional SR algorithms require multiple LR images of the same scene to generate a HR image by integrating all the information from different images [2, 3]. However, registration at sub-pixel accuracy is indispensible in order to perform SR successfully. Another type of SR algorithms requires single LR image as input [4, 5, 6]. These reconstruction based methods often use some heuristics or specific interpolation functions. The performance of these techniques degrades especially when the magnification factor becomes large.

In recent years, learning based approaches for image SR have received a lot of attention in which patterns of the images from the training set are explored. Freeman *et al.* [7] proposed an example-based method by predicting the HR images from the LR images using Markov Random Field (MRF) computed by belief propagation. Yang *et al.* [8] solved the SR problem from the perspective of compressive sensing, which ensures that under mild conditions the sparse representation of a HR image can be recovered from the downsampled signal.

In this paper, we tackle the SR problem using a learning based approach. Our SR algorithm is based on the extreme learning machine (ELM) [9]. The focus is to recover the HF components of the HR image efficiently and accurately. In the training step, features are extracted from the initially interpolated LR images (*e.g.*, using bicubic interpolation) and a model that maps the interpolated images to the HF components from the HR images is learned. Given a test LR image, we first interpolate the image. Then the HF components are estimated using the model learned during the training. By combining the interpolated image and the HF components, a final HR image is generated faithfully with sufficient details.

The remainder of this paper is organized as follows. In Section 2 ELM is briefly introduced. The proposed SR algorithm is described in Section 3. Section 4 shows the experimental results and their comparison to the *state-of-the-art* methods. Finally Section 5 concludes the paper.

2. EXTREME LEARNING MACHINE

ELM was initially developed for single-hidden-layer feedforward neural networks (SLFNs) [10]. One of the major merits of ELM is that the hidden layer needs not to be tuned. The output function of ELM is given by

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\boldsymbol{\beta} \tag{1}$$

where $\boldsymbol{\beta} = [\beta_1, \beta_2, ..., \beta_L]^T$ is a vector consisting of the output weights between the hidden layer and the output node. $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), ...h_L(\mathbf{x})]^T$ is the output of the hidden

layer given input x. Function $h(\mathbf{x})$ maps the original input data space to the L-dimensional feature space.

According to [9], ELM does not only aim at reaching the minimum training error but also the smallest norm of the output weights, which would yield a better generalization performance. Thus, in ELM the following quantities are minimized

$$\text{minimize} \begin{cases} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|^2 \\ \|\boldsymbol{\beta}\| \end{cases} \tag{2}$$

where $\mathbf{T}$ contains the training target value and $\mathbf{H}$ is the the hidden-layer output matrix

$$\mathbf{H} = \begin{bmatrix} h_1(x_1) & \cdots & h_L(x_1) \\ \vdots & & \vdots \\ h_1(x_N) & \cdots & h_L(x_N) \end{bmatrix} \tag{3}$$

In the implementation, the minimal norm least square method was used instead of standard optimization method [10]. One advantage of ELM is that in the training process the hassle of parameter tuning is avoided. The generalization performance of ELM is not sensitive to the number of hidden nodes as tested in [9]. In addition, ELM has very fast learning speed. These merits make ELM user-friendly and efficient.

3. TECHNICAL APPROACH

The proposed algorithm consists of two steps: training and testing. Figure 1 gives an overview of the proposed method. Similar to [8] we apply our method on the luminance channel only since humans are more sensitive to luminance changes. For the chrominance channels bicubic interpolation is applied.

In the training process, a number of HR images are used. The HR image I_{HR} is first blurred and downsampled by a factor of k. The downsampled image is then interpolated by a basic interpolation method (bicubic interploation in this paper) with a magnification factor of k. This step yields an initially upscaled image I_0 with the same size as I_{HR}. The HF components I_{HF} are obtained by

$$I_{HF} = I_{HR} - I_0 \tag{4}$$

Simple features from I_0 are extracted and the feature vectors fed to ELM for training consist of two components: pixel intensity values from local image patches and 1st and 2nd order derivative magnitudes.

At each pixel location (i, j) of I_0, a local patch $P_{i,j}$ of size $m \times m$ centered at (i, j) is extracted. This image patch is then reshaped into a row vector $\mathbf{p}_{(i,j)}$ of size m^2. Thus, the information about local pixel intensity values is encoded.

In order to account for the directional change in pixel intensity values, we calculate the 1st order derivatives in the horizontal and vertical directions. In addition, 2nd order derivatives are calculated to capture the rate of change in the 1st order derivatives. For each pixel 5 derivative values are obtained ($\frac{\partial I_0}{\partial \mathbf{x}}$, $\frac{\partial I_0}{\partial \mathbf{y}}$, $\frac{\partial^2 I_0}{\partial \mathbf{x}^2}$, $\frac{\partial^2 I_0}{\partial \mathbf{y}^2}$, $\frac{\partial^2 I_0}{\partial \mathbf{x}\partial \mathbf{y}}$).

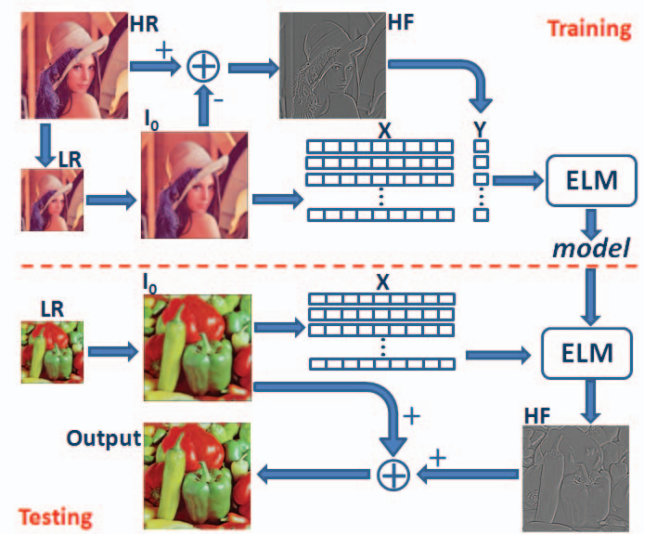

Fig. 1. The system diagram of the proposed super-resolution algorithm. LR image is obtained by blurring and downsampling the HR image. I_0 is the initially interpolated image. In the training process, feature vectors from I_0 (X) and the target values (Y) from HF (the high-frequency components obtained by subtracting I_0 from the HR image) are sent to ELM to generate a model. In the testing process, HF is predicted by ELM using trained model and the ouput is the combination of the predicted HF image and the initially interpolated image.

To calculate the 1st and 2nd order derivatives we adopt the method in [11] which is more accurate than common routine by taking difference between the adjacent pixels. Mathematically, here the derivative is formulated as an optimization of the rotation-invariance of the gradient operator. In the discrete implementation, to design a filter of length L, the error functions for the 1st and 2nd order derivatives are given by

$$E(\vec{p}, \vec{d_1}) = \frac{\left|jwF_s\vec{p} - F_a\vec{d_1}\right|^2}{|F_s\vec{p}|^2} \tag{5}$$

$$E(\vec{d_2}) = \left|j^2w^2F_s\vec{p} - F_a\vec{d_2}\right|^2 \tag{6}$$

where $\vec{p}$ is a defined parameter vector of length $\frac{L+1}{2}$ containing the independent prefilter samples. $\vec{d_1}$ contains the independent derivative kernel samples of length $\frac{L-1}{2}$. $\vec{d_2}$ contains one half of the full filters taps. F_s and F_a are matrices containing the real and imaginary components of the discrete Fourier basis. For details please refer to [11]. In our case we use the 5-tap filter. The computed 1st and 2nd order derivative values are padded to form a row vector $\mathbf{d}_{(i,j)}$.

Combining all the features together, we now have the feature vector at (i, j) as $\mathbf{v}_{(i,j)} = [\mathbf{p}_{(i,j)}, \mathbf{d}_{(i,j)}]$. The length of this feature vector is $m^2 + 5$. For the corresponding target value, we take the pixel value $i_{(i,j)}$ from I_{HF}. For each training image, the pixels are traversed in a raster scan manner.

The instances $[\mathbf{v}_{(i,j)}, i_{(i,j)}]$ from all of the HR training images are stacked together as input to ELM. After training, a model is generated that describes the mapping from the initially interpolated image to the HF image.

To super-resolve a LR image I_{LR}, the same initial interpolation is applied, generating a base image I_0. At each pixel position (i, j) in I_0, we extract the same features as we did in the training step. With the input feature vectors and the trained model, the predicted value $i_{(i,j)}$ is obtained. After going through every pixel in I_0, we then have the HF components I_{HF}. The final output I_{HR} is constructed by combining I_0 and I_{HF} together as shown in Figure 1.

4. EXPERIMENTS

In the experiment, we use 20 hidden neurons and sigmoid function as the activation function in ELM (the results are not sensitive to the number of hidden neurons as tested in our experiments). The input attributes in the feature vectors are normalized to $[-1, 1]$. Eight 512×512 sized HR images from USC_SIPI image database (http://sipi.usc.edu/database/) are used for training. We use a 5×5 Gaussian blur function with standard deviation of 1 to preprocess the HR images before downsampling. For testing, we use 25 various images from the morgueFile online archive different from the training images (http://morguefile.com, same collection of images were used in [5]). The HR images are of the size 512×512. Experiments are conducted with magnification factors of 2 and 4. The corresponding sizes of the LR images are 256×256 and 128×128. The size of the local image patch is 3×3.

We compare our method to three *state-of-the-art* methods: iterative curve based interpolation (ICBI) [5], kernel regression based method (KR) [6] and sparse representation based method (SP) [8]. The implementations of these methods are from the authors' websites. The default parameters and settings of these methods are used in our experiments. Figure 2 shows some sample results of the four methods (**Electronic version is recommended for better comparison**).

As can be seen from the results, ICBI and KR are not able to recover the HF components and the generated images suffer from blurriness at both 2x and 4x magnification. KR tends to over-smooth the images especially in the texture-rich regions. SP and our methods produces more details on the super-resolved images. However compared to SP, the results by our method are visually superior. At 2x, our method produce HR images that have vivid details and are very close to the ground truth. Even at 4x in which case the resolution of the inputs is very low, our method is still able to achieve good performance.

To measure the performance quantitatively, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [12] are calculated as shown in Table 1. At 2x the numerical scores of our method are better than all the other methods. When the magnification factor becomes 4, SP and our method both offer competitive results compared to ICBI and KR. Note that the above performance of our method is achieved by using very simple image features and the calculation involves very small feature vectors of length 14 (pixel intensity values 3×3+derivative magnitudes 5). The proposed algorithm is less complicated as compared to the other methods.

Method	ICBI [5]	KR [6]	SP [8]	Proposed
PSNR (2x)	35.03	34.64	35.95	36.74
PSNR (4x)	33.77	32.55	33.94	34.02
SSIM (2x)	0.9227	0.9125	0.9628	0.9750
SSIM (4x)	0.8520	0.7749	0.8804	0.8680

Table 1. Average PSNR and SSIM scores for different super-resolution methods (2x and 4x).

Method	ICBI [5]	KR [6]	SP [8]	Proposed
Time (2x)	3.23	21.10	727.34	3.71
Time (4x)	3.52	18.97	720.82	3.74

Table 2. Average time (in seconds) to super-resolve an image for different super-resolution methods (2x and 4x).

We also computed the average time to super-resolve an image with different magnification factors. The programs were executed on a desktop with Intel Core2 2.4 GHz CPU and 3 GB of RAM. Here we do not compare the training time since this is a one-time offline process, although in the training process ELM converges very fast (within 20 seconds). From Table 2 we can see that the running time of all the methods are not sensitive to the magnification factors. Our method without code optimization is very fast with different magnification factors. The running time of ICBI is close to our method but the output quality is less satisfactory. KR processes images at a moderate rate without competitive performance. Although SP achieves similar performance to our method at 4x, it takes much more time to generate the output. Due to the efficiency of our method, many real-time applications are possible.

5. CONCLUSIONS

In this paper, an efficient algorithm for image super-resolution based on extreme learning machine (ELM) is proposed. During the training process, simple features including pixel intensity values in a local image patch and the 1st and 2nd order derivative magnitudes are extracted. The target value is the high-frequency components obtained by subtracting the initially interpolated low-resolution image from the high-resolution one. ELM then learns a model that maps the interpolated image to the high-frequency components. Given a low-resolution image, same features are extracted from the interpolated image. By applying the trained model, ELM is able to predict the high-frequency components. The final output is the combination of the interpolated image and the high-frequency components. Compared to the *state-of-the-art* methods, our method achieves high performance in both subjective and quantitative evaluations, and less complicated. Furthermore, the computation of our method is very efficient. Involving a more comprehensive dataset and more sophisticated image features would be promising for better

Fig. 2. From left to right: results by ICBI [5], results by KR [6], results by SP [8], results by the proposed method, original images. From top to down: super-resolution at 2x and 4x.

performance and these aspects will be investigated in our future work.

Acknowledgment This work was supported in part by NSF grant 0905671.

6. REFERENCES

[1] S. Biswas, G. Aggarwal, and P.J. Flynn, "Pose-robust recognition of low-resolution face images," in *Proc. CVPR*, 2011.

[2] R. Tsai and T. Huang, "Multi-frame image restoration and registration," *Advances in Computer Vision and Image Processing*, 1984.

[3] S. Farsiu, M.D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE TIP*, 2004.

[4] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Proc. CVPR*, 2008.

[5] A. Giachetti and N. Asuni, "Real time artifact-free image upscaling," *IEEE TIP*, 2011.

[6] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE TIP*, 2007.

[7] W.T. Freeman, E.C. Pasztor, and O.T. Carmichael, "Learning low-level vision," *IJCV*, 2000.

[8] J. Yang, J. Wright, T.S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE TIP*, 2010.

[9] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE SMCB*, 2011.

[10] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proc. IJCNN*, 2004.

[11] H. Farid and E.P. Simoncelli, "Differentiation of discrete multidimensional signals," *IEEE TIP*, 2004.

[12] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE TIP*, 2004.