

Image Enlargement as an Edge Estimation

Chi-kin Chow and Hung-tat Tsui
Visual Signal Processing and Communications Laboratory
Department of Electronic Engineering
The Chinese University of Hong Kong
Hong Kong SAR, China
E-mail: {ckchow1, httsui}@ee.cuhk.edu.hk

Abstract - A robust image enlargement algorithm is presented in this paper. We formulate the image enlargement process as an edge information estimation process. In order to achieve a higher resolution, we first perform Pixel Duplication [1] on the target image to form an initial high resolution image. Then the edge details of the enlarged image are estimated by using a novel neural network called "Agent Swarm Regression Network ASRN", which is trained by a set of low resolution (LR) / high resolution (HR) image patch pairs. Two benchmark images were used to verify the performance of the proposed algorithm. The results show that the enlarged images by the proposed algorithm are sharper than those by the conventional methods.

I. INTRODUCTION

A numbers of approaches have been used to enlarge an image for various applications. Generally, there is a tradeoff between the computation speed and the visual artifacts introduced. One of the typical approaches is called *Bicubic interpolation*. It estimates the intensity at a pixel in the target image by the weighted sum of its 16 neighbor pixels in the source image.

Besides the mathematical approaches, some researchers focus on enlarging an image by using a set of examples [2, 3]. They have some success on predicting the missing edges in an enlarged image. A significant contribution to the example-based super resolution algorithm was made by Freeman, Jones and Paszter [2]. They used the normalized high frequency training patches, as well as the spatial consistency constraint to enlarge a wide class of images. One important point to be noted in [2] is that they do not directly generalize the collected training set. Instead, their algorithm requires about 100,000 patches to handle a wide range of image enlargement cases.

In this paper, a new type of neural network is proposed to estimate the edge information of the target image. It can give sharper enlarged image than those generated by other standard approaches. The proposed algorithm was found to be robust for enlarging the image whose texture is the same, or somewhat different from the training set. Moreover, a relatively smaller training set is required to perform good image enlargement quality.

This paper is organized as follows. In section 2, the problem of image enlargement is modeled as an edge estimation process, and the neural network approach is suggested to solve it. A multi-agent system (MAS) based radial basis function (RBF) neural network called "Agent Swarm Regression Network ASRN" is proposed together with its training algorithm are discussed in Section 3. A low-resolution benchmark image will be enlarged in order to demonstrate the performance of the proposed algorithm in Section 4. A conclusion is drawn in Section 5.

II. THE PROPOSED EDGE ESTIMATOR *

The problem of edge estimation is raised when an image is scaled up to a larger size. Some researchers suggest using the neighboring pixels to interpolate in the enlarged image. *Pixel Duplication* or *Bicubic interpolation* are typical examples. It is obvious that the scaled-images are blurred due to the lack of high-resolution edge information. Therefore, the image enlargement process can be modeled as an edge recovery process.

Unless the higher resolution of an image is provided, the HR edges can never be determined from the corresponding LR image. Therefore, instead of recovering the HR edges, we would like to estimate it based on a set of HR/LR image pairs. Besides the correlation approach, neural network based function module called "*Neural Edge Estimator NEE*" is introduced so as to give a better generalization on the edge estimation.

NEE Training Phase – Similar to other artificial neural network applications, the proposed RBF network must be trained before undergoing any estimation or regression. In the training phase, a set of HR/LR image patch pairs are provided to the network in order to teach it how to predict the enlarged edge information from the LR patches. We randomly sub-divide the given HR training image into a set of small image patches, and then transform into the training set for the NEE. Consequently, the network will be well-trained through the training algorithm discussed in Section 3. Figure 1 gives a summary of the procedure that transform a given HR image patch into a training sample.

1. Given a HR image patch \mathbf{H} (fig. 2c).
2. The corresponding LR image patch \mathbf{L} (fig. 2a) can be generated by *averaging the intensities*.
3. Normalize the contrast of \mathbf{L} to form the input of training sample L_{CN} (fig. 2b) by using the equation:

$$L_{CN}(x, y) = \frac{L(x, y) - \min_{u, v} L(u, v)}{\max_{u, v} L(u, v) - \min_{u, v} L(u, v)} \quad \text{Eq. (1)}$$
4. Apply *Pixel Duplication* on \mathbf{L} is form an analytic HR image patch \mathbf{H}_a .
5. Determine the HR edge patch, i.e. $\mathbf{E} = \mathbf{H} - \mathbf{H}_a$.
6. The output of the training sample \mathbf{O} is defined as:

$$O(x, y) = \frac{E(x, y)}{\max_{u, v} L(u, v) - \min_{u, v} L(u, v)} \quad \text{Eq. (2)}$$

Figure 1 Steps of generating NEE experience

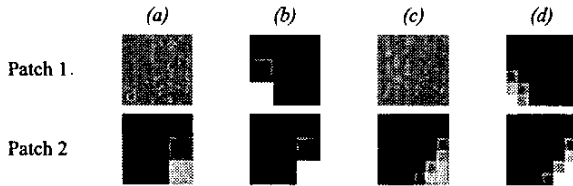


Figure 2. Examples of NEE Experience Generation. (a) LR image patches. (b) The image patches in (a) with normalized contrast. (c) The corresponding HR image patches of (a). (d) The image patches in (c) with normalized contrast.

Image Enlargement Phase—To enlarge a target LR image, we divide it into a set of LR image patches \mathbf{I} that are traversed in raster-scan order. Then we apply the enlargement process discussed in below to form the corresponding HR image patches \mathbf{I}_H . The resultant HR image can be constructed by integrating the \mathbf{I}_H with the same order in LR image patches division. For each \mathbf{I} , it is initially doubled by pixel duplication to form the analytic enlarged image \mathbf{I}_L . On the other hand, \mathbf{I} is transformed into the network input vectors \mathbf{T}_L by the Eq. (1). After passing through the trained network, the estimated edge information with normalized contrast \mathbf{T}_H is generated at the network output. The resultant enlarged image patch \mathbf{I}_H is constructed by simply adding the scaled edge information \mathbf{T}_H onto \mathbf{I}_L .

$$\text{i.e. } \mathbf{I}_H = \mathbf{I}_L + \mathbf{T}_H \times (\text{Max } \mathbf{T}_L - \text{Min } \mathbf{T}_L) \quad \text{Eq. (3)}$$

III. AGENT SWARM REGRESSION NETWORK

Various definitions from different disciplines have been proposed for the term Multi-Agent System (MAS) [4-6]. As seen from Distributed Artificial Intelligent (DAI), a multi-agent system is a loosely coupled network of problem-solver entities that work together to find answers to problems that are beyond the individual capabilities or knowledge of each entity. More recently, the term multi-agent system has been

given a more general meaning. It is now used for all types of systems composed of multiple autonomous components showing the following characteristics:

- Each agent has incomplete capabilities to solve a problem.
- There is no global system control.
- Data is decentralized.
- Computation is asynchronous.

In general, the behavior of an MAS agent can be decomposed into 3 stages. Initially, an agent collects information from the environment and its neighbor(s) in order to conclude its local environment observation (LEO). Based on the LEO, the agent will draw a decision from its decision-making unit, which benefits to its current situation. Afterward, the agent executes the decision by adjusting its internal parameters. Since the agent swarm keeps interact with the environment, there is no beginning nor terminating stages. In addition, as the swarm undergoes those 3 stages in parallel instead of sequentially, a more complicated swarm behavior is constructed from a set of simple rules.

A. Parameter of ASRN agent

For a MAS-based application, the features of an agent are described by its abilities. In this application, the ASRN agent consists of 8 parameters. 4 of them are adjustable during the iterations while the remaining 4 parameters are predefined at the beginning of swarm evolution. The follows are the details of the agent parameters.

- 1) *Magnitude M* : The magnitude of an agent is its weight on the whole regression network. The magnitudes of the whole swarm are assigned as one initially.
- 2) *Reduced Magnitude ΔM* : Quantity of magnitude to be reduced at the current evolution
- 3) *Position vector μ* : Position of the training sample.
- 4) *Desired Output Y* : Desired Output of the training sample.
- 5) *Variance σ* : This parameter describes the correlation among the agent with its neighbor(s). To simplify the training process, all neurons of the network are set to be the same value. Instead of predefining the value by the user, we suggest the variance based on the following equation:

$$\sigma = p^{\frac{1}{D}} (-8 \ln 0.8)^{\frac{1}{2}} \quad \text{Eq. (4)}$$

where p is the swarm size and D is the swarm dimension

- 6) *Local Environment Observation (LEO) O* : The regression value at μ .
- 7) *Neighbor Linkage N* : It is a P by 1 vector, i.e. $\mathbf{N}_i = [n_{i,1} \ n_{i,2} \ \dots \ n_{i,P}]$ where P is the swarm size. $n_{i,j}$ equals to 1 if Agent \mathbf{A}_j is the neighbor of \mathbf{A}_i . Otherwise, $n_{i,j}$ is set to be zero. Therefore, $\mathbf{N}_i \cdot \mathbf{N}_i^T$ represents the number of neighbors that \mathbf{A}_i communicates with.

8) *Magnitude Decay Rate* α : This parameter controls the magnitude decrement rate of an agent. Instead of a constant decay rate along the iterations, the rate of decrement is changed adaptively. The adaptive strategy will be discussed at the Section III D

B. Neighbor of ASRN agent

Different from DAI that every agent can collect information from all the others, the MAS agents can only communicate with a limited number of agents inside the swarm. We named the agents that can be communicated by agent A_i as the neighbors of A_i . It is obvious that as the number of neighbors increases, more information can be collected. Hence a more accurate decision can be drawn. On the other hand, more time is needed to analyze the collected information. Therefore, for a robust MAS, the agents should achieve the global objective by communicating with a minimum number of neighbors. In the case of ASRN, the neighbors of an agent A_i are defined as follows:

1. Calculate the influence of agent A_j on A_i : $\{G_{ij}\}$, i.e. $G_{ij} = \exp(-\|\mu_i - \mu_j\| / 2\sigma^2)$ for all $j \neq i$.
2. Sort $\{G_{ij}\}$ in the descending order of magnitude.
3. Select a minimum of p agents with the largest influence as the neighbors of A_i such that the total influence of them is greater than:

$$\beta \sum_j^p G_{ij}$$

where β is the decentralized accuracy with $0 \leq \beta \leq 1$

C. LEO of ASRN agent

Theoretically, the observed regression is defined as:

$$O_i = M_i + \sum_{j=1}^{N_i} M_{I_{ij}} e^{-\frac{|R - \beta_{I_{ij}}|}{2\sigma^2}} \quad \text{Eq. (5)}$$

where I_{ij} is the agent index of the j^{th} neighbor of the i^{th} agent. However, since the positions and the variances of all agents are fixed, the LEO can be rewritten as a linear equation in which the training process is accelerated.

D. Decision Making Unit of ASRN agent

Based on the constraints of an ideal network described at Section 3, the behavior of an agent can be formulated as below such that the swarm will converge to an optimal network.

1. Determine the regression error E_i , i.e. $E_i = Y_i - O_i$
2. If Maximum Error Threshold (E_{max}) $\geq |E_i|$ then
 - $\alpha_i \leftarrow 2\alpha_i$; $\Delta M_i \leftarrow \alpha_i \times M_i$
 - Else
 - $\alpha_i \leftarrow 0.5\alpha_i$; $\Delta M_i \leftarrow M_i - E_i$
 - Endif

E. Action Unit of ASRN agent

After determining the quantity of magnitude reduced, the agent take an action by simply decreasing its magnitude, i.e. $M_i \leftarrow M_i - \Delta M_i$.

IV. EXPERIMENTAL RESULT

Next, we evaluate the effectiveness of the proposed algorithm by estimating the higher resolution version of a 256×256 benchmark image shown in fig. 4a. A total of 3000 NEEs are extracted from the training image (fig. 3) in order to train the ASRN. The NEEs are generated by the 3×3 image patches and aim at scale the image size in twice. The trained ASRN is then applied on the LR version images at fig 4a to form the resultant HR image (fig. 4b). Moreover, the enlarged images by the *Bicubic interpolation BI* and *Pixel Duplication PD* are listed in fig. 4c and 4d respectively, in order to provide a performance comparison. The results show that edges estimated by the proposed algorithm are shaper than that generated by *BI* and *PD*.

TABLE I
PSNR OF THE EXPERIMENTAL RESULT

	Pepper	Lotus
Pixel Duplication	23.858	18.911
Bicubic Interpolation	27.991	21.354
The Proposed Method	29.042	24.251

V. CONCLUSION

In this paper, we propose to formulate the image enlargement process as a superposition of an analytic HR image with the estimated HR edges. The main focus is on the estimation of the HR edges by using the guidance of the LR/HR image patch examples. Different from the recent example-based approach that searches the nearest edge patch exhaustively from the database, the proposed algorithm models the edge enlargement behavior by using of a newly proposed MAS-based RBF network - *ASRN*. The experimental results show that the proposed algorithm can generate the enlarged image with the edges sharper than that generated by *BI*. Further, the texture of the training image can be somewhat different from the target one. Due to the generalization ability of the ASRN, the proposed algorithm requires only a small number of NEE to construct the edge estimator that estimate the HR edge information of an image patch not in the NEE set.

REFERENCES

- [1] W.K. Pratt, *Digital Image Processing*, Joh Wiley & Sons, Inc., 1991.
- [2] Freeman, W.T.; Jones, T.R.; Pasztor, E.C., "Example - based super - resolution", *IEEE Trans. on Computer Graphics and Applications*, Volume: 22 Issue: 2, March - April 2002 Page(s): 56 - 65.
- [3] Simon Baker and Takeo Kanade, "Limits on Super-Resolution and How to Break Them", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Volume: 24, No. 9, September 2002 Page(s): 1167 -1183.

- [4] Xi Cheng; Yi-Bin Hou; "Modeling the multi-agent system", Proceedings of the International Conference on Machine Learning and Cybernetics 2002, Volume: 1, 4-5 Nov. 2002, Pages: 4 – 7.
- [5] Ying Li; Bao Liu; "The advances of multi-agent system (MAS)", Proceedings of the 3rd World Congress on Intelligent Control and Automation, 2000, Volume: 1, 28 June-2 July 2000, Pages: 246 – 249.

- [6] Loureiro da Costa, A.; Bittencour, G.; "Dynamic social knowledge: a cognitive multi-agent system cooperation strategy", Proceedings of the International Conference on Multi Agent Systems, 1998, 3-7 July 1998, Pages: 415 – 416.

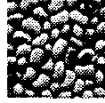
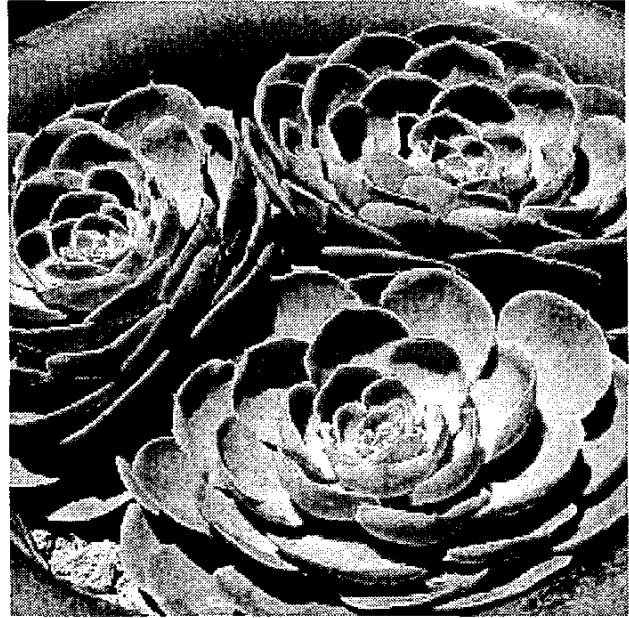
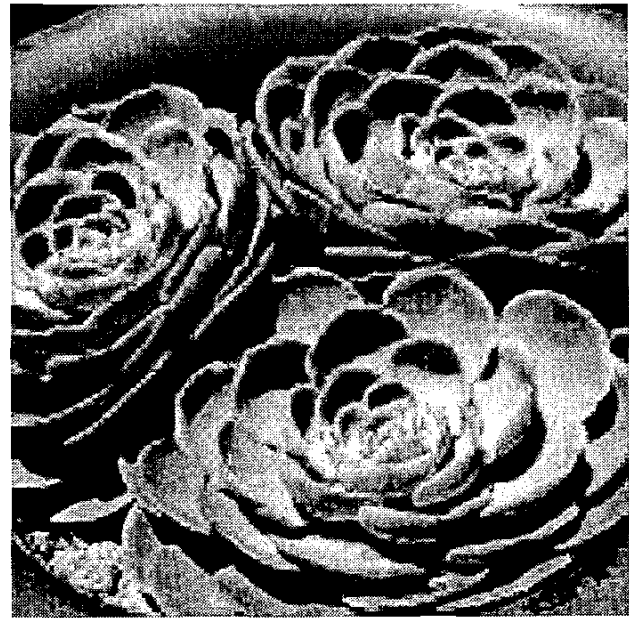


Figure 3 The training image in Section 4 – a 320x320 gray scale image.



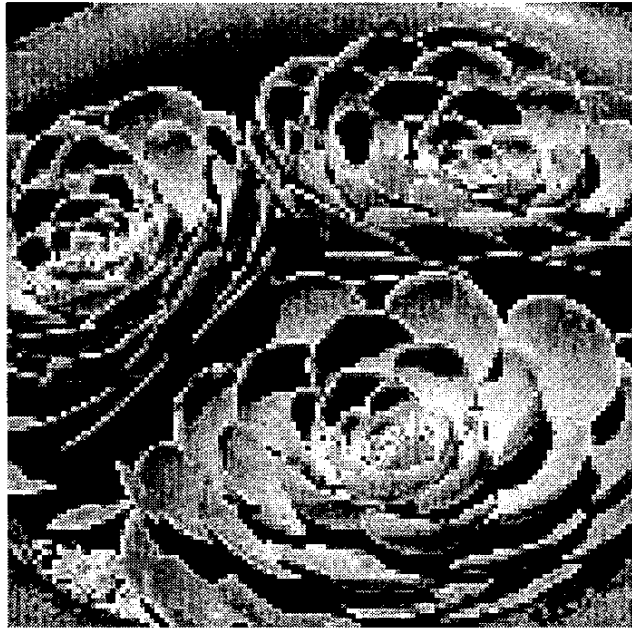
(a)



(b)



(c)



(d)

Figure 4 Experimental Results of Section 4. (a) The original HR image. (b) The enlarged image by the proposed algorithm. (c) The enlarged image by Bicubic interpolation. (d) The enlarged image by Pixel Duplication