Int. J. Electron. Commun. (AEÜ) 74 (2017) 88–93

Contents lists available at ScienceDirect

International Journal of Electronics and Communications (AEÜ)

journal homepage: www.elsevier.com/locate/aeue

Regular paper

Single face hallucination via local neighbor patches

Mei Gong *, Ke Wang

School of Software Engineering, Chengdu University of Information Technology, Chengdu 610225, China

ARTICLE INFO

Article history:
Received 18 November 2015
Accepted 31 January 2017

Keywords:
Face hallucination
Local window
Neighbor patch
Similarity measurement

ABSTRACT

Based on learning neighborhood patches a new single face hallucination method is proposed in this paper. In the proposed method, each input low-resolution (LR) position-patch and all patches in a local window centered at the same position of training images are used to hallucinate a high-resolution (HR) face patch, meanwhile two local similarity measurements between each input LR patch and all local LR and HR neighborhood patches of training images are computed to constrain the hallucination. Additionally, a residue image is estimated for the further improvement of the reconstructed result. Experimental results show that the proposed method can obtain superior or competitive results.

1. Introduction

HR images, especially interested HR face images are very useful in video surveillance, face recognition and many other applications. However, due to the long distance or device limitations etc, the captured interested face images are often LR. In order to obtain more details or useful information from a given LR image, learning-based single image super-resolution(SR) is proposed [1,2], and learning-based face image SR or hallucination algorithm is first proposed by Baker et al. [3]. Chang et al. [2] used locally linear embedding [4] technique to do SR reconstruction. Wang and Tang [5] used Principal Component Analysis technique to hallucinate face. Ma et al. [6] proposed a hallucinating face approach by position-patch. Jung et al. [7] used convex optimization to stably hallucinate face without considering the locality. Jiang et al. [8] and Li et al. [9] respectively used a locality constraint and a local linear filter to hallucinate face.

Position information of a patch is useful for face hallucination [6–8], but different persons may have some subtle differences or characteristics, and if the alignment between face images is inaccurate, the same position-patches in different face images may not always correctly reveal the same structures or features. Thus, in this paper, for each input patch we firstly collect all LR/HR patches in a local neighborhood of the same position from each training LR/HR image, then compute two local similarity measurements to constrain the hallucinated weight. Additionally, a residue image is estimated for the further improvement of the reconstructed result.

2. The proposed method

2.1. Descriptions

Given an input LR face image X_t, a HR training set $Y^H = \{Y_1, Y_2, \ldots, Y_M\}$, and the corresponding LR training set $X^L = \{X_1, X_2, \ldots, X_M\}$, where M is the number of training face images. Divide X_t into N small overlapping patches, the size of each patch is $s \times s$, and each patch is represented by its feature vector, the feature patch matrix with $\mathrm{R} \times C$ size is $L_t = \{x_t(i,j)|i = 1, 2, \ldots, R \quad j = 1, 2, \ldots, C\}$, where (i,j) is the position of each patch in L_t. Similarly, each LR face image in the training set is also divided into N small overlapping patches with $s \times s$ patch size, and the feature patch set is $\{x_m(i,j)|i = 1, 2, \ldots, R \quad j = 1, 2, \ldots, C\}$, correspondingly, each HR training face image is also divided into N overlapping patches with $rs \times rs$ patch size, and the feature patch set is $\{y_m(i,j)|i = 1, 2, \ldots, R \quad j = 1, 2, \ldots, C\}$, where r is a resolution enhancement factor, $m = 1, 2, \ldots, M$.

Inspired by [2,10], we also use a feature vector to represent each LR patch, and the feature vector consists of the first-order gradients, second-order gradients, and the sum of luminance difference between the center pixel and all local neighbor pixels in the patch. For each HR training patch, the mean value is subtracted from each pixel, and all luminance deviations are concatenated to represent the HR patch.

2.2. Similarity measurement in the local neighborhood of a position-patch

Different from the traditional position-based methods in [6–8], the proposed method not only considers the position-patch, but

* Corresponding author.
E-mail addresses: flyhightop@163.com (M. Gong), wangke@cuit.edu.cn (K. Wang).

http://dx.doi.org/10.1016/j.aeue.2017.01.022
1434-8411/

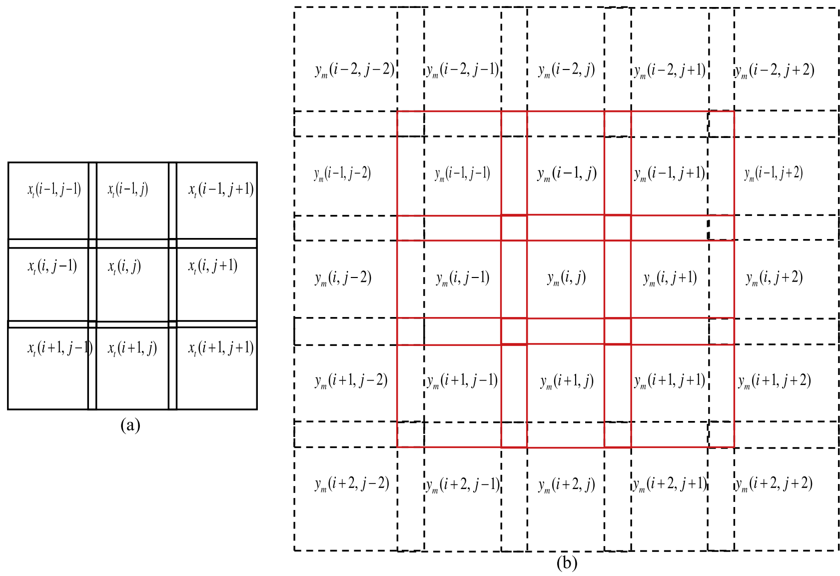

Fig. 1. (a) LR local neighborhood of $x_t(i,j)$. (b) The neighborhood of all HR neighbor patches in each HR training image.

also employs its neighbor patches in a local window, simultaneously uses two local distances to measure the similarities between LR and LR patches, and between LR and HR patches in the local window.

For each $x_t(i,j)$, we collect all LR and HR patches in a 3×3 window centered at position (i,j) from each training image, and the local LR and HR patch sets are represented as $L_l(i,j) = \{x_m(p,q)|(p,q) \in \Omega(i,j)\}_{m=1,2,\ldots,M}$ and $L_h(i,j) = \{y_m(p,q)|(p,q) \in \Omega(i,j)\}_{m=1,2,\ldots,M}$, where $\Omega(i,j) = \{(i-1,j-1),(i-1,j),(i-1,j+1),(i,j-1),(i,j),(i,j+1),(i+1,j-1),(i+1,j),(i+1,j+1)\}$ is the patch indexes in the local window. Thus, the first similarity measurement $d^k_{l\_l}(i,j)$ between $x_t(i,j)$ and all $x_m(p,q) \in L_l(i,j)$ is calculated as:

$$d^k_{l\_l}(i,j) = \|x_t(i,j) - x_m(p,q)\|_2^2, \quad 1 \leqslant k \leqslant 9M \tag{1}$$

When calculating the second distance $d^k_{l\_h}(i,j)$ to measure the similarity between $x_t(i,j)$ and all $y_m(p,q) \in L_h(i,j)$, because of different resolution and size of LR and HR patches, and inspired by [9], we firstly calculate the LR patch similarity $S_t(i,j)$ in a 3×3 local neighborhood of $x_t(i,j)$ from the input LR image, and measure the HR patch similarity $S_m(p,q)$ in a 3×3 local neighborhood centered at position (p,q) of each HR training image respectively, then use $S_t(i,j)$ and $S_m(p,q)$ to calculate $d^k_{l\_h}(i,j)$. The LR neighbor patches of $x_t(i,j)$ in the local window of the input LR image and the corresponding neighborhood of all HR neighbor patches in each HR training image are as shown in Fig. 1.

As shown in Fig. 1(a), a LR similarity $s^n_t(i,j)$ between $x_t(i,j)$ and all patches $\{x_t(p,q)|(p,q) \in \Omega(i,j)\}$ in the 3×3 window is represented as:

$$s^n_t(i,j) = \|x_t(i,j) - x_t(p,q)\|_2^2, \ (p,q) \in \Omega(i,j) \ , 1 \leqslant n \leqslant 9 \tag{2}$$

Then combine all $s^n_t(i,j)$ in the local window as $S_t(i,j)$:

$$S_t(i,j) = [s^1_t(i,j), s^2_t(i,j), \ldots, s^9_t(i,j)] \tag{3}$$

From Fig. 1(b), when $p = i-1$, $q = j-1$, the patch indexes in a 3×3 window centered at position (p,q) is $\Omega(p,q) = \{(i-2,j-2),(i-2,j-1),(i-2,j),(i-1,j-2),(i-1,j-1),(i-1,j),(i,j-2),(i,j-1),(i,j)\}$, and a HR similarity $s^n_{\mathrm{m}}(p,q)$ between $y_m(p,q)$ and all $\{y_{\mathrm{m}}(p_h,q_h)|(p_h,q_h) \in \Omega(p,q)\}$ is:

$$s^n_{\mathrm{m}}(p,q) = \|y_m(p,q) - y_m(p_h,q_h)\|_2^2, \quad (p_h,q_h) \in \Omega(p,q), \quad 1 \leqslant \mathrm{n} \leqslant 9 \tag{4}$$

Combine all $s^n_m(p,q)$ as the HR patch similarity measurement $S_m(p,q)$:

$$S_m(p,q) = [s^1_m(p,q), s^2_m(p,q), \ldots, s^9_m(p,q)] \tag{5}$$

Then the second similarity measurement $d^k_{l\_h}(i,j)$ is:

$$d^k_{l\_h}(i,j) = \|S_t(i,j) - S_m(p,q)\|_2^2, \quad 1 \leqslant k \leqslant 9M \tag{6}$$

2.3. Face hallucination via position-patch and its local neighbor patches

For each $x_t(i,j)$, we firstly collect its local LR/HR training set $\{L_l(i,j), L_h(i,j)\}$, if $x_t(i,j)$ is a border patch in the patch matrix, $\{L_l(i,j), L_h(i,j)\}$ is only composed of the same position-patches, otherwise $\{L_l(i,j), L_h(i,j)\}$ is composed of all training patches in a 3×3 window centered at position (i,j). Then based on learning-patches and inspiration from [2,6–8,11], we incorporate two distance measurements $d_{l\_l}(i,j)$ and $d_{l\_h}(i,j)$ to constrain the hallucination. Consequently, our reconstructed function is:

$$w(i,j) = \arg\min_{w(i,j)} \left\{ \left\| x_t(i,j) - \sum_{x_m(p,q) \in L_l(i,j)} w_k(i,j) x_m(p,q) \right\|_2^2 + \lambda_1 \sum_{k=1}^{9M} \left[d^k_{l\_l}(i,j) w_k(i,j) \right]^2 + \lambda_2 \sum_{k=1}^{9M} \left[d^k_{l\_h}(i,j) w_k(i,j) \right]^2 \right\} \tag{7}$$

where $w(i,j) = [w_1(i,j), w_2(i,j), \ldots, w_{9M}(i,j)]$ is the weight, and $\sum_{k=1}^{9M} w_k(i,j) = 1$, $d_{l\_l}(i,j) = [d^1_{l\_l}(i,j), d^2_{l\_l}(i,j), \ldots, d^{9M}_{l\_l}(i,j)]$, $d_{l\_h}(i,j) = [d^1_{l\_h}(i,j), d^2_{l\_h}(i,j), \ldots, d^{9M}_{l\_h}(i,j)]$. Accordingly, the initial HR patch $y^0_t(i,j)$ can be reconstructed by:

$$y^0_t(i,j) = w(i,j) L_h(i,j) \tag{8}$$

2.4. Residue image compensation

For each $y^0_t(i,j)$, we search K nearest HR neighbors from the training space as the HR neighbors set $N_h = \{y_1, y_2, \ldots, y_K\}$, and minimize the error e_h as in [2]: