Spatially Adaptive Block-Based Super-Resolution

Heng Su, Liang Tang, Ying Wu, *Senior Member, IEEE*, Daniel Tretter, and Jie Zhou, *Senior Member, IEEE*

***Abstract*—Super-resolution technology provides an effective way to increase image resolution by incorporating additional information from successive input images or training samples. Various super-resolution algorithms have been proposed based on different assumptions, and their relative performances can differ in regions of different characteristics within a single image. Based on this observation, an adaptive algorithm is proposed in this paper to integrate a higher level image classification task and a lower level super-resolution process, in which we incorporate reconstruction-based super-resolution algorithms, single-image enhancement, and image/video classification into a single comprehensive framework. The target high-resolution image plane is divided into adaptive-sized blocks, and different suitable super-resolution algorithms are automatically selected for the blocks. Then, a deblocking process is applied to reduce block edge artifacts. A new benchmark is also utilized to measure the performance of super-resolution algorithms. Experimental results with real-life videos indicate encouraging improvements with our method.**

***Index Terms*—Block based, motion registration error, spatially adaptive framework, super-resolution, super-resolution benchmark.**

I. Introduction

THE GOAL of super-resolution image reconstruction technology is to generate high-resolution (HR) images from input low-resolution (LR) images. After this was first addressed in 1984 [1], super-resolution technologies have been extensively studied and widely used in satellite imaging, medical image processing, traffic surveillance, video compression, video printing, and other applications.

In general, most contemporary super-resolution algorithms can be classified into two categories: reconstruction-based algorithms and learning-based algorithms. Reconstruction-based algorithms usually combine information from a set of successive LR frames of the same scene to generate one or several HR images, which is an ill-posed estimation problem, mathematically. The basic idea of reconstruction-based super-resolution is to exploit additional information from successive LR frames with subpixel displacements and then to synthesize an HR image or a sequence. Early super-resolution methods solve the problem in the frequency domain but are usually restricted to global translational motion and linear space-invariant blur [1], [2]. Most contemporary algorithms solve the super-resolution problem in the spatial domain. Iterative back-projection [3], [4] algorithms estimate the HR image by iteratively back projecting the error between simulated LR images and the observed ones. Maximum *a posteriori* (MAP) [5]–[7] approaches adopt the prior probability of target HR images to stabilize the solution space under a Bayesian framework. Projection on convex sets (POCS) [8], [9] tends to consider the solution as an element on a convex set defined by the input LR images. The wavelet-based super-resolution algorithm [10] has been proposed. However, these approaches are computationally demanding. Noniterative interpolation-based super-resolution [11] is introduced to remove the frequency aliasing in the reconstruction process. However, this method is relatively sensitive to registration errors.

On the other hand, learning-based super-resolution algorithms [12]–[18] usually extract redundant high-frequency image information from training samples containing known HR components, rather than using successive LR image frames. One crucial problem in learning-based super-resolution algorithms is the representation of the high-frequency component of an HR image.

Every algorithm (referred to as candidate algorithms in this paper) has its own assumptions and, hence, is restricted to specific kinds of image regions. For example, MAP methods achieve better results where there are suitable prior knowledge (such as face-image super-resolution), and iterative methods fit image regions with small registration error; otherwise, error could be accumulated during iteration [19]. Therefore, when both successive LR images and training sample data are given, one may expect that better results could be obtained by combining different super-resolution algorithms into a single framework and apply different algorithms to different regions of a scene in order to improve the robustness and the quality of results. That is one of the motivations of this paper.

In addition, single-frame image enhancement and interpolation is a mature topic in image processing. Heuristic image enhancement technology is also referred to as super-resolution [20]. The work of Cha and Kim [21] was a recent approach that introduces an edge-forming algorithm based on a partial differential equation model. It can also be applied in SR processing where there are no accurate corresponding LR image patches.

In this paper, a practical extendable block-based super-resolution algorithm combination framework is proposed. The target

Manuscript received February 06, 2011; revised July 01, 2011; accepted August 16, 2011. Date of publication September 01, 2011; date of current version February 17, 2012. This work was supported in part by the HP Labs China; by the Computational Vision Group of Northwestern University; by the Natural Science Foundation of China under Grant 60721003, Grant 60875017, Grant 61020106004, and Grant 61021063; by the Science and Technology Support Program of China under Grant 2009BAH40B03; by the National Science Foundation under Grant IIS-0347877 and Grant IIS-0916607; and by the US Army Research Laboratory and the U.S. Army Research Office under Grant ARO W911NF-08-1-0504. An earlier version of this paper was presented at the IEEE International Conference on Image Processing, San Diego, CA, October 12–15, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xilin Chen.

H. Su and J. Zhou are with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: su-h02@mails.tsinghua.edu.cn; jzhou@tsinghua.edu.cn).

L. Tang is with No. 45 Research Institute, China Electronics Technology Group Corporation, Beijing 100176, China (e-mail: tangliang@45inst.com).

Y. Wu is with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208 USA (e-mail: yingwu@eecs.northwestern.edu).

D. Tretter is with HP Labs Palo Alto, Hewlett–Packard Company, Palo Alto, CA 94304 USA (e-mail: dan.tretter@hp.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2011.2166971

image is divided into adaptive-sized blocks, which are classified into categories by their features according to the corresponding LR video patches; then, different candidate super-resolution algorithms and single-image enhancement algorithms are applied to blocks in different categories. The purpose of the framework is to increase the robustness of the method and to make the best use of each candidate algorithm according to the characteristics of each image block. A deblocking process is also applied to reduce block edge effects.

The framework can be implemented in various ways for different practical situations. We also propose an implementation of the framework in this paper to solve super-resolution problems for image sequences with large local motion and to reduce computational cost. A two-stage adaptive block generation and classification process is used to classify the image blocks and applies a suitable candidate algorithm for each block in the implementation. The first stage of the process is designed as rule based, and the second is learning based; therefore, both predefined empirical rules and information extracted from the training set are utilized in block classification to increase the robustness. A significant advantage of our framework implementation over algorithms such as the confidence map [22], [23] is that the proposed algorithm has more flexibility; we can analyze image blocks not only by local registration error but also by the magnitude of motion fields, smoothness, texture, and all kinds of features of images/videos.

The rest of this paper is arranged as follows. In the next section, we introduce the formulation of the proposed framework, particularly some notations concerning the block analysis process. An overview of both the framework and the framework implementation are provided in Section III. The technical details are presented in Sections IV and V: Section IV shows the process of block analysis, and Section V explains the proposed spatially adaptive super-resolution algorithm method. In Section VI, experimental results are reported to verify the effectiveness of the proposed algorithm, and we also propose a new benchmark scheme to measure the average performance of super-resolution algorithms. Finally, conclusions are made in Section VII.

II. Problem Formulation

A. Super-Resolution Image Reconstruction Formulation

Suppose that there are altogether n LR images. The super-resolution imaging model can be written as [24]

$$L_k = DB_k^{(2)} M_k B_k^{(1)} H + N_k, \qquad k = 1, \ldots, n \tag{1}$$

where each of the observed LR images L_k is assumed to be generated from the original HR image H through a sequence of transformation or effects: the atmosphere blur effect $B_k^{(1)}$, followed by the motion transformation M_k; the imaging blur effect $B_k^{(2)}$ caused by the camera imaging system (i.e., motion blur effect and CCD blur effect); and, finally, the downsampled transformation D. N_k represents the additive noise to each LR image. The goal of super-resolution image reconstruction is to find an estimate $\hat{H}$ of the real HR image H.

All image variables H, L_k, and N_k in (1) are represented as column vectors composed of the pixel intensity of corresponding images in the lexicographical order, respectively; thus, the transformation or effects applied to images can be represented as matrix multiplication operations. Assuming that the LR images are of size $n_x \times n_y$ and that the super-resolution magnification factor is $r_x \times r_y$, then the sizes of vectors and transform matrices are fixed: L_k and N_k are $(n_x n_y) \times 1$; H and $\hat{H}$ are $(r_x n_x r_y n_y) \times 1$; $B_k^{(1)}$, $B_k^{(2)}$, and M_k are $(r_x n_x r_y n_y) \times (r_x n_x r_y n_y)$; and D is $(n_x n_y) \times (r_x n_x r_y n_y)$.

In most reconstruction-based super-resolution algorithms, the first step is to estimate matrices $B_k^{(1)}$, $B_k^{(2)}$, and M_k. Nevertheless, motion registration proves to be a difficult task for practical captured image sequences, particularly the ones containing relatively large local motion between frames. A lot of super-resolution research assumes global motion such as perspective motion/affine motion [6] or even pure translational motion, which limits the application of such algorithms. One motivation of this paper is to develop a practical super-resolution method robust to motion registration errors.

B. Block Analysis Formulation

The target image is divided into several adaptive-sized nonoverlapping blocks in the proposed framework. Each block is represented as a rectangular region within the target scene. In this section, a mathematical formulation related to blocks in the framework is proposed to make the description in the following sections clearer.

The set of all blocks is represented as $\mathcal{B} = \{b_i | i = 1, \ldots, m\}$, where m and b_i denote the number of blocks and the ith block, respectively. According to the nonoverlapping restriction, the following formulation holds:

$$\bigcup_i b_i = b_h, \qquad i = 1, \ldots, m \tag{2}$$

$$b_i \cap b_j = \varnothing \quad \forall i, j \tag{3}$$

where b_h denotes the block of the whole target image H.

Given the definition of blocks, its necessary to restrict (1) within a certain block to analyze super-resolution performance in the corresponding block area. For simplicity, we write (1) as

$$L_k = W_k H + N_k, k = 1 \ldots n \tag{4}$$

where $W_k = DB_k^{(2)} M_k B_k^{(1)}$. The token $b_i^{(k)}$ is defined as the corresponding area of b_i in the kth LR image as follows:

$$b_i^{(k)} = \{(x, y) | \exists (p, q) \in b_i, \text{s.t. } W_k(x, y; p, q) \neq 0\} \tag{5}$$

where $W_k(x, y; p, q)$ denotes the matrix entry with respect to pixel (p, q) in the HR grid and pixel (x, y) in the LR grid.

An image data vector being indexed by a block or an area extracts the corresponding image data within the block or area in lexicographical order. For example, $H(b_i)$ and $L_k(b_i^{(k)})$ represent the lexicographically ordered image data of the HR image H within block b_i and the LR image L_k within area $b_i^{(k)}$, respectively. $(W_k)_i$ is defined as the submatrix of W_k restricted on b_i as follows:

$$(W_k)_i = W_k\left(b_i^{(k)}; b_i\right) \tag{6}$$

where $W_k(\mathcal{R}; \mathcal{C})$ extracts the submatrix composed by elements with a row index in $\mathcal{R}$ and a column index in $\mathcal{C}$ of the orig-

inal matrix W_k, where $\mathcal{R}$ and $\mathcal{C}$ are both sets of indexes. Thus, $L_k(b_i^{(k)})$ and product $(W_k)_i H(b_i)$ are vectors of the same size.

We further normalize the values of each row in $(W_k)_i$ and denote the resulting matrix as $(W_k)_i'$, as given in

$$(W_k)_i'(x;y) = (W_k)_i(x;y)\frac{\sum_p W_k(x';p)}{\sum_p (W_k)_i(x;p)} \tag{7}$$

for all row index x and column index y of $(W_k)_i$, where x' denotes the row index in W_k corresponding to the xth row of $(W_k)_i$. In most cases, matrices D, $B_k^{(2)}$, $B_k^{(1)}$, and M_k satisfy

$$\begin{aligned}\sum_p D(x;p) &= \sum_p B_k^{(2)}(x;p)\\ &= \sum_p M_k(x;p)\\ &= \sum_p B_k^{(1)}(x;p)\\ &= 1 \quad \forall x.\end{aligned} \tag{8}$$

Thus, we can easily obtain

$$\sum_p W_k(x';p) = 1 \tag{9}$$

Therefore, when (8) holds, (7) can be rewritten as

$$(W_k)_i'(x;y) = \frac{(W_k)_i(x;y)}{\sum_p (W_k)_i(x;p)}. \tag{10}$$

$(W_k)_i'$ is called the normalized matrix of $(W_k)_i$, thus, we have

$$\begin{aligned}\sum_q (W_k)_i'(x;q) &= \sum_q (W_k)_i(x;q)\frac{\sum_p W_k(x';p)}{\sum_p (W_k)_i(x;p)}\\ &= \sum_p W_k(x';p) = 1.\end{aligned} \tag{11}$$

III. Framework Overview

A. Proposed Framework

After registering input LR images into the HR grid [i.e., estimating motion between frames, the blur effect matrix, and thus matrix W_k in (4)], the target HR image plane is divided into adaptive-sized blocks according to their features in the block analysis step. Every block b_i is then classified and assigned a candidate super-resolution algorithm ω_i^*, which is the candidate algorithm that minimizes the expectation of the estimation error given b_i, as shown in

$$\omega_i^* = \arg\min_{\omega} E_\omega[e|b_i] \tag{12}$$

where e is the error between the estimated HR image and the ground-truth image and $E_\omega[e]$ represents the expectation of estimation error when adopting the candidate algorithm ω. Thus, if we apply algorithm ω_i^* to every block b_i, the overall expectation of estimation error $E^*[e]$ is minimized, and the performance is improved. For all candidate algorithm ω, we have

$$\begin{aligned}E_\omega[e] &= \sum_{b_i} E_\omega[e|b_i]p(b_i)\\ &\geq \sum_{b_i} E_{\omega_i^*}[e|b_i]p(b_i) = E^*[e].\end{aligned} \tag{13}$$

In the synthesis step, the super-resolution algorithm ω_i^* is applied to b_i to generate the HR image result $\hat{H}$ as follows:

$$\hat{H}(b_i) = \hat{H}_{\omega_i^*}(b_i) \tag{14}$$

where $\hat{H}_\omega$ represents the super-resolution result of the candidate algorithm ω.

Finally, before the HR image result is produced, a deblocking process is introduced to reduce the block edge effect between neighboring blocks with different category labels.

There are different ways to generate the block set, to extract block features, and to classify the blocks; thus, the process of the framework is not fixed. The framework can be implemented under different situations by adopting specified image feature extraction, block segmentation, block classification, and parameter selection algorithms. For example, block classification can be designed as a rule-based process when an explicit difference between block categories exists, or as a learning-based one when sufficient training samples are provided.

B. Implementation of the Framework

In this section, we present an implementation of the introduced framework. The main purposes of the implementation are two: to improve super-resolution image reconstruction performance under poor motion registration accuracy and to colligate the advantages of candidate super-resolution algorithms. The flowchart of the proposed framework implementation is shown in Fig. 1.

The whole target HR image area is analyzed and divided into adaptive-sized blocks based on the content of the image and registration accuracy within the blocks. We use a two-stage hierachical classification method. The first stage is rule based, and the second is learning-based.

In the first stage, the blocks are classified into three patterns by predefined rules, including flat image blocks, mismatched image blocks, and well-registered image blocks. For the first two block patterns, an appropriate single-frame processing method is chosen and applied to blocks in each pattern directly. In the second stage for the last block pattern of well-registered image blocks, a learning-based classifier is employed to further assign a candidate super-resolution algorithm or a single-image enhancement method for each block.

Image denoising and single-image enhancement approaches are also incorporated in the framework implementation. This reduces the computational cost of the approach within flat blocks without affecting HR image quality and improves the visual result within mismatched blocks.

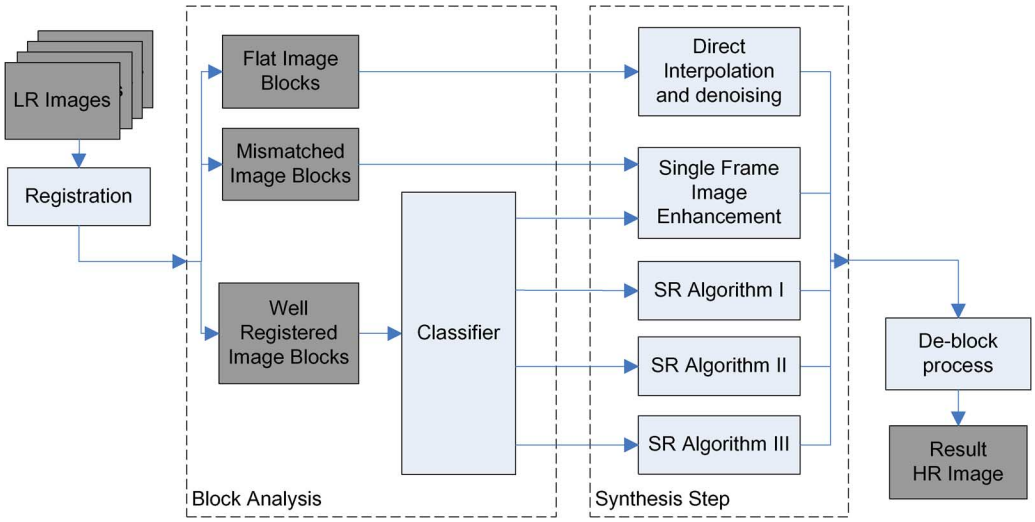


Fig. 1. Sample implementation of the proposed super-resolution framework.

IV. Block Analysis

A. Adaptive Block Generation

Most reconstruction-based super-resolution algorithms consist of two steps: the registration step and the synthesis step. In the registration step, transformation parameters are estimated. In the synthesis step, registration results are used to estimate the target HR image. Many of the algorithms mentioned focus on the synthesis step. However, researchers have realized the significance of the registration step [25]. Unfortunately, limited to current computer vision and image processing technology, it is impossible to register LR images very accurately in all cases, particularly in the presence of large local motion.

A variety of work has been done to solve the registration problem. An intuitive way is to increase the accuracy of motion estimation in super-resolution [26]. Milanfar [5] states that using L1 norm instead of L2 norm increases robustness in the presence of registration errors. Joint estimation methods [25], [27], [28] iterate to get better registration results but sometimes suffer from convergence problems. Block-matching 3-D filtering [29], probabilistic motion estimation [30], [31], and steering kernel regression [32], [33] are introduced in super-resolution to increase the robustness. Another kind of algorithm, known as the confidence map [22] or channel adaptive regularization [23], tries to reduce the contribution of LR images or parts with large registration error in the synthesis. In [34], the target image area is divided into adaptive-sized blocks according to the local texture and motion compensation error to eliminate the influence of inaccurate optical flow.

The proposed adaptive-sized block analysis overcomes the registration error problem in a different way. There are reasons for dividing the target HR image into adaptive-sized blocks, rather than regions of arbitrary shapes, as most region-based SR algorithms do. First, a coarse-to-fine block analysis algorithm is computationally cheaper than a pixelwise region segmentation algorithm for the same purpose. Second, our method is more efficient in removing spatial noise in the image, for small image areas or single pixels that are mislabeled will be eliminated by the block in the proposed algorithm. Moreover, block-based super-resolution has natural facility for compressed video processing, for prevailing video compression algorithms are based on fixed- or adaptive-sized blocks [35].

First, the whole HR image plane is divided into $p \times q$ square blocks of the same size m_{init} as the initial block division set: $\mathcal{B} = \{b_i | i = 1, \ldots, p \times q\}$. Then, the features of each block b_i are analyzed to refine the block division set and to label the block pattern.

A structure matrix is used to examine the smoothness around a pixel, where the structure matrix for a single pixel $I(x, y)$ in image I is defined as

$$S(x, y) = \nabla I(x, y) \cdot \nabla I(x, y)^T. \tag{15}$$

Thus, we can define the structure matrix of block b_i as

$$S_i = \frac{1}{n_i} \sum_{(x,y)\in b_i} S(x, y) = \frac{1}{n_i} \sum_{(x,y)\in b_i} \nabla \tilde{H}(x, y) \cdot \nabla \tilde{H}(x, y)^T \tag{16}$$

where n_i represents the number of pixels in block b_i. Notice that the original HR image H is unknown; therefore, the guess $\tilde{H}$ of the original H is used instead. $\tilde{H}$ can be generated by directly interpolating the input LR image or further by denoising the interpolated image. This substitution makes (16) inaccurate. That is one of the reasons we use adaptive-sized blocks, which reduces the spatial noise by averaging the structure matrix within a block. There is also a similar situation in (18).

In the experiment, we use the first-order Sobel operator to extract the image gradient, and the LR image is directly interpolated to estimate $\tilde{H}$. Note that if some denoising process is adopted, the smoothness is dependent on the parameters of the denoising process; therefore, direct interpolation is better for a general algorithm. On the other hand, if the proposed algorithm is applied in some specific situations (e.g., the noise model of the camera is known), a well-designed denoising process will be helpful to improve the result.

The eigenvalues $\lambda_1^{(i)}$ and $\lambda_2^{(i)}$ of matrix S_i are a measurement of gradient strength in two perpendicular directions. The larger eigenvalue corresponds to the direction with the stronger gradient, whereas the smaller one corresponds to the direction with

the weaker gradient. The smoothness σ_i of block b_i is defined as

$$\sigma_i = \left|\lambda_1^{(i)}\right| + \left|\lambda_2^{(i)}\right|. \tag{17}$$

The corresponding area of b_i in the kth LR image $b_i^{(k)}$ [i.e., defined in (5)] is defined to be mismatched when the measured pixel values in L_k differ from the expected values by more than a certain threshold l, i.e., $b_i^{(k)}$ is mismatched only if

$$\left\|(W_k)_i'\tilde{H}(b_i) - L_k\left(b_i^{(k)}\right)\right\| > l, \qquad k = 1 \ldots n \tag{18}$$

where $(W_k)_i'$ is defined as the normalized matrix of $(W_k)_i$ in (7). We use the normalized matrix instead of the original one so that the values in $(W_k)_i'\tilde{H}(b_i)$ and $L_k(b_i^{(k)})$ are comparable. Then, the match measurement τ_i of block b_i can be defined as

$$\tau_i = \frac{1}{n - n_a}\left(\left|\left\{b_i^{(k)} | b_i^{(k)} \text{ is mismatched} \quad \forall k\right\}\right| - n_a\right) \tag{19}$$

where $|\mathcal{A}|$ denotes the number of elements in $\mathcal{A}$ when $\mathcal{A}$ is a set. In order to prevent the block classification being biased toward mismatched blocks when the motion is very large, we discard all the absent frames, in which all blocks are mismatched. In (19), n_a represents the number of the absent frames, and n_a is usually zero in practice.

The blocks are classified into three patterns on the first level (see Fig. 1): flat blocks, mismatched blocks, and registered blocks. Algorithm 1 describes the details of the block analysis process.

Algorithm 1: Adaptive Block Generation

Input: $\tilde{H}$, L_k, and W_k.

Algorithm parameters: initial block size m_{init}, threshold σ_L, τ_L, τ_H, and minimum block size m.

1. Initialize block division set $\mathcal{B}$ with square blocks of the same size m_{init} : $\mathcal{B} = \{b_i | i = 1, \ldots, p \times q\}$;
2. Go to 6 if there are no unlabeled blocks in $\mathcal{B}$;
3. Pick an unlabeled block b_i from $\mathcal{B}$;
4. If $\sigma_i < \sigma_L$, label b_i as flat block; or if $\tau_i \geq \tau_H$, label b_i as mismatched block; or if $\tau_i < \tau_L$, or size of b_i is smaller than m, label b_i as registered block. If none of the conditions above are satisfied, eliminate b_i from $\mathcal{B}$, and break b_i into 2×2 square blocks of the same size, add all the 4 smaller blocks into $\mathcal{B}$;
5. Go to 2;
6. Output $\mathcal{B}$ with the labels of the blocks.

Therefore, the maximum breaking times n_{break} for an arbitrary initial block is

$$n_{\text{break}} = \left\lfloor \frac{\ln(m_{\text{init}} - m)}{\ln 2} \right\rfloor \tag{20}$$

and the minimum possible block size $m_{\min}$ in the output $\mathcal{B}$ is

$$m_{\min} = \frac{m_{\text{init}}}{2^{n_{\text{break}}}}. \tag{21}$$

Flat blocks contain the least information in the image. Subpixel displacements provide no more information for the flat region, except for noise statistics. Thus, only the direct interpolation and denoising process is applied for those blocks to reduce the computational cost. The mismatched blocks contain meaningful information, but the information in the different input LR images does not seem to correspond to the same image data; therefore, the single-frame image enhancement approach should be applied to increase the image quality. Only well-registered blocks deserve SR algorithms to exploit reliable additional information from input LR frames. In the proposed framework implementation, well-registered blocks are further classified into second-level categories, where each category corresponds to one candidate super-resolution algorithm, which will be discussed in Section V.

B. Deblocking Process

Visual discordance may appear at the edges between different categories of blocks enhanced by different algorithms. A deblocking process is necessary to obtain smooth transition across the edge after the synthesis step of our framework.

To accomplish this, the blocks along the edge area are dilated, resulting in an overlapping region along the edge. Suppose that the width of the overlapping area is $2r_0$, and $I_1(x, y)$ and $I_2(x, y)$ denote the result image of the two blocks in the overlapping area. Thus, the final result image $I(x, y)$ in the overlapping area is obtained by

$$I(x, y) = I_1(x, y)f(r) + I_2(x, y)f(-r) \tag{22}$$

where r represents the distance between (x, y) and the edge, and $f(r)$ represents the combine function and satisfies $f(-r_0) = 1$, $f(r_0) = 0$, and $f(r) = 1 - f(-r)$.

Our goal is to find the optimal function $f^*(r)$ that gets the best combined result, which means that $f^*(r)$ should not vary severely with r. Therefore, we get

$$\begin{aligned} f^*(r) &= \arg\min \int_{-r_0}^{r_0} |f'(r)|^2 \, dr \\ &= \frac{r_0 - r}{2r_0}. \end{aligned} \tag{23}$$

Thus, $f^*(r)$ is a linear function. In Fig. 2,we show that both visual consistency and image fidelity [in peak signal-to-noise ratio (PSNR) (31)] are improved by adopting the proposed deblocking process. A brief proof that the proposed deblocking process improves the image fidelity (in PSNR) is shown in Appendix A.

Note that adjacent blocks tend to be in the same category (see Figs. 5 and 8 in the following section) because adjacent blocks tend to have similar statistics and thus are more likely to be classified into the same category. Therefore, the deblocking process only needs to be applied in a relatively small portion of the whole image area. This is why a simple linear combination deblocking process is sufficient in the proposed algorithm.

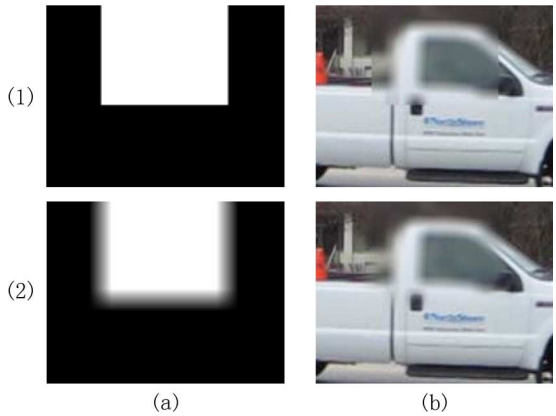


Fig. 2. Deblocking image result. Column (a) Combine function. Column (b) Image results synthesized by components with different resolutions: the blurred image [within the white box in (a)] and the original image [the dark area in (a)]. Row (1) Without deblocking, PSNR $= 25.4253$ dB. Row (2) With deblocking, $r_0 = 8$, and PSNR $= 25.8223$ dB. The block artifact occurs in (1b), which is removed in (2b).

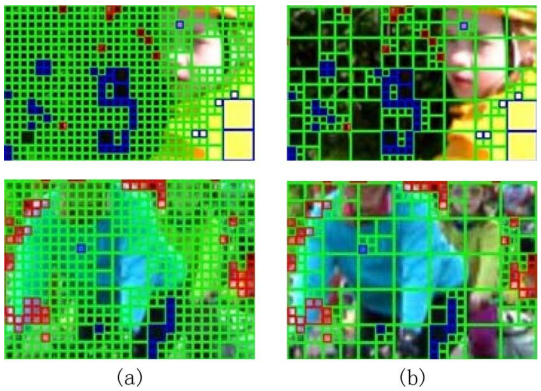


Fig. 3. Block reduction algorithm.Column (a) Image patches with oversegmented blocks after block generation. Column (b) Block results after a block reduction algorithm. Blue blocks represents flat ones, red for mismatched ones, and green for registered ones.

V. Spatially Adaptive Super-Resolution

A. Recursive Block Reduction

According to the block generation algorithm proposed in Section IV-A, for an arbitrary block b_i, if $\tau_L \leq \tau_i \leq \tau_H$ and the match measurement values τ of all the subblocks of b_i are similar, which is often the case, block b_i is likely to be divided into the minimum block size $m_{\min}$. In other words, the proposed block generation algorithm tends to generate a lot of blocks of the same category with the minimum block size $m_{\min}$, particularly for well-registered block category [see Fig. 3(a)]. This leads to oversegmentation of the target image plane and makes the block generation result highly dependent on the minimum block size $m_{\min}$. In some cases, this has little impact on the final HR image. For example, in the proposed framework implementation, direct interpolation with a denoising process is directly assigned for flat block image area, even if flat blocks are oversegmented, the computational cost and the performance of the process will not be affected very much.

However, oversegmentation can bring negative influence to learning-based candidate-algorithm selection scheme, in both the training stage and the testing stage. Oversegmentation increases the number of samples, which causes a higher computational cost. Moreover, block generation result dependent on $m_{\min}$ means that the block division depends on algorithm parameters and thus depends on the scale information of the target image, which decreases the robustness of the learning-based process. It should also be noticed that oversegmentation is inevitable, for the features (such as τ_i and σ_i) of the subblocks of a block, which are necessary for analyzing the details of the block, can only be retrieved by breaking the block into subblocks, and the process leads to oversegmentation. Thus, a block reduction process is introduced after the block generation process to ensure the robustness of the framework when learning-based block classification algorithms are involved.

The recursive block reduction process merges oversegmented subblocks of the same category into larger blocks, which can be regarded as an inverse procedure of block generation to some extent. Despite that the process is recursive, its computationally effective because the algorithm works on the image-block level rather than the pixel level. We start from blocks with the minimum block size $m_{\min}$ and hierachically merge the blocks into larger ones. The details of the block reduction algorithm are shown in Algorithm 2. Notice that only adjacent blocks, which are generated by breaking a single larger block, can be merged in the algorithm.

Algorithm 2: Recursive Block Reduction

Input: Block division result $\mathcal{B}$ of the block generation process with the label of each block.

Algorithm parameters: Maximum merging times n_{merge}.

1. Scan all blocks in $\mathcal{B}$. Denote every group of 2×2 adjacent blocks with the same category and with size m_{min} as p_i , where i represents the index of block groups that satisfy the above constraints. Define $\mathcal{P} = \{p_i\}$;
2. Pick an unprocessed element p_i in $\mathcal{P}$. Process p_i as the following;
3. Let $k = 1$;
4. Merge the 2×2 blocks in p_i into a larger block b_i^k. Add b_i^k into $\mathcal{B}$ with the same category as blocks in p_i, and remove all blocks in p_i from $\mathcal{B}$;
5. $k \leftarrow k + 1$;
6. If $k > n_{\text{merge}}$, go to 8;
7. Search that if there are $3(= 2 \times 2 - 1)$ other blocks in $\mathcal{B}$ with the same type as b_i^{k-1} and adjacent to b_i^{k-1}. If not, go to 8. If so, merge b_i^{k-1} and the three blocks into b_i^k and remove them from $\mathcal{B}$. Add b_i^k into $\mathcal{B}$ with the same category as block b_i^{k-1}. Go to 5;
8. If there are still unprocessed elements in $\mathcal{P}$, go to 2, or else output $\mathcal{B}$ with the labels of the blocks;

It is easy to prove that the output $\mathcal{B}$ of the block reduction algorithm does not contain any blocks smaller than $m_{\min}2^{n_{\text{merge}}}$ with the same category that can be merged. Typically, we can set $n_{\text{merge}} = n_{\text{break}}$; therefore, the largest size of merged blocks is $m_{\min}2^{n_{\text{break}}} = m_{\text{init}}$.

If the blocks generated from the block generation algorithm is stored in a certain order (e.g., in a first-in–first-out order),

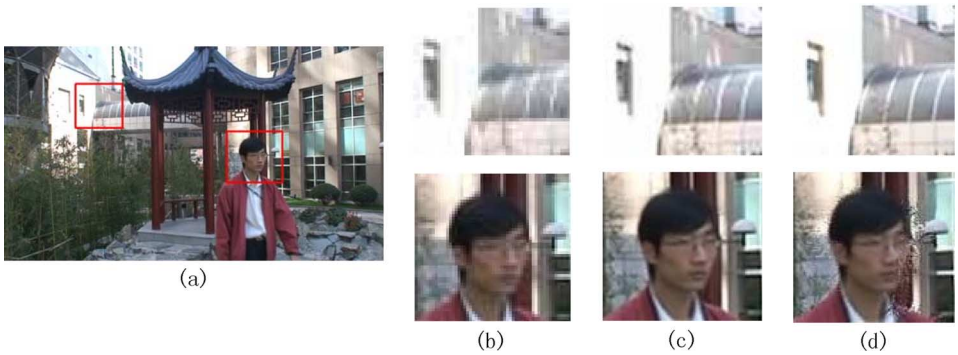


Fig. 4. Comparison of the MAP method [5] and POCS method [8]. Column (a) Ground-truth HR image with two image patches in red boxes. Column (b). LR images of the selected image patches. Column (c) MAP [5] result. Column (d) POCS [8] result. It is shown that POCS gets better result in the background building with more accurate motion, whereas MAP has better performance around the human face area with less motion registration accuracy.

the block reduction algorithm can be computationally effective because we do not need to search for all blocks in $\mathcal{B}$ for adjacent blocks in step 7.

The selection of parameters m_{init} and m [see (20)] (or n_{merge} and $m_{\min}$) controls the sizes of the generated blocks. The blocks are used in a classification process, and both the training and testing sets adopt the same parameters. Thus, if sufficient training samples are provided, the parameters are robust to the classifier. Typically, if there are sufficient training samples, m_{init} should be large to capture the overall content of the local image, and m should be relatively small to generate detailed blocks if necessary. However, if m_{init} is large, a huge number of training samples are needed, which are usually impossible in practice. Therefore, m_{init} should be selected according to the number of the training samples. In our experiments, the number of the training samples is in the magnitude of 10^4 (see Section VI). Therefore, we choose $m_{\text{init}} = 24$ and $m = 6$ in all our experiments.

The results of block reduction are shown in Fig. 3.

B. Learning-Based Combination

As mentioned above, different candidate algorithms are applied to different registered blocks based on their categories, as determined by the second classification stage. The idea is based on the fact that each candidate super-resolution algorithm has specific assumptions about input data and different characteristics. It is shown in [19] that an iterative method with relatively bad registration accuracy gets worse results than algorithms without iteration, for the motion registration error accumulates during the iteration process. On the other hand, with accurate enough motion registration results, iterative algorithms improve HR quality by refining the results.

In Fig. 4, we also illustrate the comparative results of two iterative algorithms. The POCS method [8] gets better results in the background with relatively more accurate motion registration over the MAP method [5], whereas the MAP method [5] provides an obviously better reconstruction result with much less artifacts around the human face where motion registration is not accurate enough. The best candidate algorithm for each well-registered block in two sample super-resolution tasks is shown in Fig. 5.

We propose a learning-based combination scheme for different super-resolution algorithms in this subsection. The well-registered blocks, which are generated in the first-level rule-based classification process discussed in Section IV-A and merged using the block reduction algorithm (Algorithm 2), are further classified into second-level categories. Each category corresponds to one candidate super-resolution algorithm, and the final result is generated by combining results from processing each block by the appropriate algorithm [see (14)]. Notice that the deblocking process described in Section IV-B also applies along the boundaries between blocks with different second-level categories.

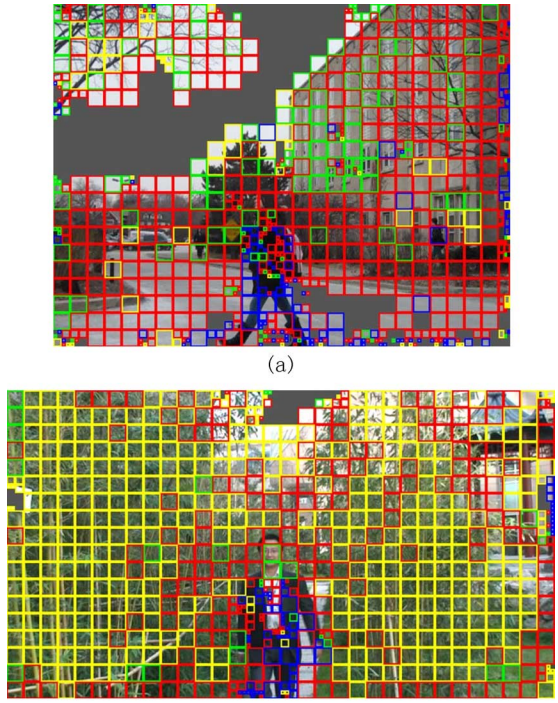


Fig. 5. Best candidate algorithms for well-registered blocks in two sample super-resolution tasks. The color of each block represents the best candidate algorithm of the corresponding block based on the PSNR criteria in (31): Red, yellow, green, and blue represent algorithms of MAP [5], POCS [8], confidence map [23], and single-image enhancement [21], respectively, and dark area within the images indicates region of flat blocks/mismatched blocks.

A single block b_i in a certain super-resolution task is considered as one sample in the learning-based classification process. The information of the sample corresponding to block b_i is mainly provided by the image content of the input LR image sequence within the area corresponding to b_i, which is

referred to as $L_k(b_i^{(k)})$. First, the LR image content is linearly transformed into the HR grid by matrix $(V_k)_i'$, where $(V_k)_i'$ is a normalized matrix defined as follows, similarly to (10):

$$(V_k)_i'(x;y) = \frac{(V_k)_i(x;y)}{\sum_p (V_k)_i(x;p)} \tag{24}$$

where

$$(V_k)_i = (W_k)_i^T. \tag{25}$$

$(V_k)_i'$ can be regarded as an approximate scaled inverse transformation of $(W_k)_i$ and projects the image content in $b_i^{(k)}$ in the kth LR image into the block grid b_i in the HR image plane. In most cases, a certain pixel in the target HR image is related to only a few pixels in a given LR image. For example, suppose a super-resolution task with a magnification factor of 2×2, a 3×3 space-invariant blur kernel, and a global translational motion. Assuming that bilinear interpolation is used in motion compensation, under some simple mathematical manipulation, we can get that one pixel in the HR image is on average only associated with approximately four pixels in any one of the LR input images according to matrix W_k. Thus, $(W_k)_i$ and $(V_k)_i'$ are sparse matrices, and the values of nondiagonal elements in $(V_k)_i'(W_k)_i$ are small compared with those of the diagonal elements in the same matrix, which makes $(V_k)_i'(W_k)_i$ an approximate diagonal matrix. The normalized $(V_k)_i'$ is adopted instead of $(V_k)_i$ because the elements in $(V_k)_i'L_k(b_i^{(k)})$ are comparable to the original image intensity value, which is easier to store and to manipulate. Notice that, although W_k is sparse, $\sum_k W_k^T W_k$ is not sparse enough; therefore, the conventional super-resolution object function, i.e.,

$$\hat{H} = \arg\min_H \left(\sum_k \|W_k H - L_k\|^2 + \|FH\|^2 \right) \tag{26}$$

where F is the regularization filter, cannot be solved directly.

The HR image estimate $\tilde{H}(b_i)$ within block b_i and the transformed motion registration error $(S_k)_i$ are adopted as the feature vector of a sample corresponding to block b_i in a super-resolution task, where $(S_k)_i$ is defined as

$$(S_k)_i \triangleq D(V_k)_i' \left[(W_k)_i' \tilde{H}(b_i) - L_k\left(b_i^{(k)}\right) \right]. \tag{27}$$

The feature vector is further downsampled into the LR grid to reduce the dimensionality. We also include the Fourier transform and the wavelet transform results of the image contents in the feature vector. Therefore, the feature vector X_i is defined as

$$X_i = \begin{bmatrix} ED\tilde{H} \\ \alpha E(S_1)_i \\ \vdots \\ \alpha E(S_n)_i \end{bmatrix} \tag{28}$$

where

$$E = \begin{bmatrix} I \\ \mathcal{F} \\ \mathcal{W} \end{bmatrix}. \tag{29}$$

$\mathcal{F}$ and $\mathcal{W}$ represent the 2-D discrete Fourier transform (2-D-DFT) matrix and the wavelet transform matrix, respectively. Notation α is a scalar to adjust the weight of the motion registration error. The feature vector (28) can be rewritten as

$$X_i = (\Lambda \otimes E) K_i \tag{30}$$

where Λ is the diagonal weight matrix $\mathrm{diag}(1, \alpha, \ldots, \alpha)$, E is the feature extraction matrix defined as (29), and K_i is the data vector $[(D\tilde{H})^T, ((S_1)_i)^T, \ldots, ((S_n)_i)^T]^T$. The operator $\otimes$ represents the Kronecker matrix product operator. Equation (30) separates the three influence parts of X_i and shows the extendability of the proposed algorithm, as one can adjust Λ to change the weight of each data block in K_i or to add/change linear filters in E to change the formation of feature vectors.

In order to make all the feature vectors X_i to have the same dimension as the ones corresponding to the blocks of size m_{init}, all the content of X_i is linearly resized into the same size as those blocks of size m_{init}. For notation simplicity, the new resized vector is still referred to as X_i in the following part of this paper.

Notice that if the kth frame is taken as the super-resolution target reference image, the corresponding $(S_k)_i$ of the feature vector is $\vec{0}$; therefore, they can be eliminated from the feature vector to reduce the dimensionality.

The label of each sample in the training set is generated by comparing the result of every candidate super-resolution algorithm with the ground-truth HR image. The index of the candidate algorithm with the best result is set as the label of the training sample. We use the well-known PSNR criteria to measure the result quality, which is defined as

$$\mathrm{PSNR}(H, I) = 10 \log_{10} \frac{P^2}{\frac{1}{n_x n_y} \sum_{i,j} (H(i,j) - I(i,j))^2} \tag{31}$$

where H and I denotes the reference image and the distorted image, respectively; $n_x \times n_y$ is the size of the image; and P represents the maximum possible value in H and I, which is equal to $255 (= 2^8 - 1)$ for 8-bit images.

The feature vectors X_i in the training and testing sets are directly selected as the input sample feature vectors of the training and testing stages of the classifiers. The classification process learns a recognition model to discriminate which candidate super-resolution algorithm works best on a given block. Note that the information in each block is simply stacked in the corresponding feature vector in (30) without heuristic filtering or predealing and that we do not know exactly which specific subset of the features is the most effective in distinguishing between the candidate algorithms. Thus, classifiers with the capability of selecting appropriate features (such as boosting [36] or random forest [37]) would achieve better performance in general. The specific selection of the classifiers in our experiment is described in Section VI-B. The performance of a classification problem also depends on the training set selected, particularly in the case where a high-dimensional feature set is used. However, because of the essential distinction between different candidate super-resolution algorithms, e.g., that mentioned in the beginning of this subsection, we will show that

the proposed algorithm is capable of improving the results with different kinds of training sets in Section VI-B.

VI. Experimental Results

Several experiments were carried out to verify the effectiveness of our algorithm. Unlike conventional super-resolution algorithms, the proposed framework implementation selects and applies suitable candidate algorithms for each part of the target image; therefore, the way to verify the effectiveness of the proposed approach should be different but compatible with the widely used PSNR-based criteria. Moreover, the performance of super-resolution and image enhancement algorithms adopting learning methods often depends on the training set selected, particularly when the sample is defined as an image sequence, rather than a single image, which increases the complexity of the training data structure. Thus, we not only show the reconstructed results of the proposed implementation on several specific LR image sequences but also introduce a benchmark scheme given a database of a lot of image sequence samples. The benchmark scheme can be applied to most super-resolution and image enhancement algorithms, i.e., both reconstruction based and learning based.

A. Experiment Setup

In the experiments, we use seven successive image frames as the input of each single super-resolution task. The middle (fourth) frame of the seven images is the super-resolution target image, and a 2×2 super-resolution process is applied on each set of the seven image frames.

The input image frames are generated from video clips captured by handheld high-quality video cameras: First, original frames are extracted from the video clips. Then, the original frames are 2×2 downsampled into LR images as input of super-resolution algorithms, whereas the original sequence is considered as target HR ground-truth images for image quality measurement (such as PSNR). Additive white Gaussian noise is added to the downsampled LR images to simulate the noise in the video capture process. All images are color images with three (RGB) channels.

The atmosphere blur matrix $B_k^{(1)}$ [see (1)] is simply set as the identity matrix I, and the imaging blur effect $(B_k^{(2)})$ is set as a 3×3 Gaussian space-invariant linear kernel with the deviation of 1 pixel wide. We tried two motion estimation algorithms to estimate the motion matrix M_k: the method of [38] and the algorithm described in [23]; the latter of which gives a better motion estimation result for our data. Thus, it is selected as the motion estimation algorithm in the experiment. The motion estimation algorithm described in [23] is a pyramid-based hierarchical one aiming to minimize the sum of squared differences (SSDs) between the compensated and the original images. The motion field, which is modeled by a set of global affine motion parameters and local parameters specifying the smoothness-constrained local additional flow field, is estimated using the image pyramid in a coarse-to-fine manner. The image pyramid used in the proposed algorithm has four levels.

The parameters of the framework implementation are set as follows. In the block analysis and the first-level rule-based block classification step, the initial block size m_{init} is 24; the minimum block size $m = 6$, which makes n_{break} equals 2; and the thresholds σ_L, τ_L, and τ_H, are set as 150, 0.7, and 1, respectively. In the second-level learning-based block classification step, three conventional super-resolution algorithms, i.e., MAP [5], POCS [8], and confidence map [23], are selected as candidate algorithms. A single-color-image enhancement algorithm [21] is selected as the single-frame enhancement algorithm in mismatched area and also as a candidate algorithm for registered blocks (see Fig. 1). $n_{\text{merge}} = n_{\text{break}} = 2$. A two-stage Haar wavelet decomposition is used to extract wavelet features of the samples. $\alpha = 0.25$. The deblocking overlap margin r_0 is

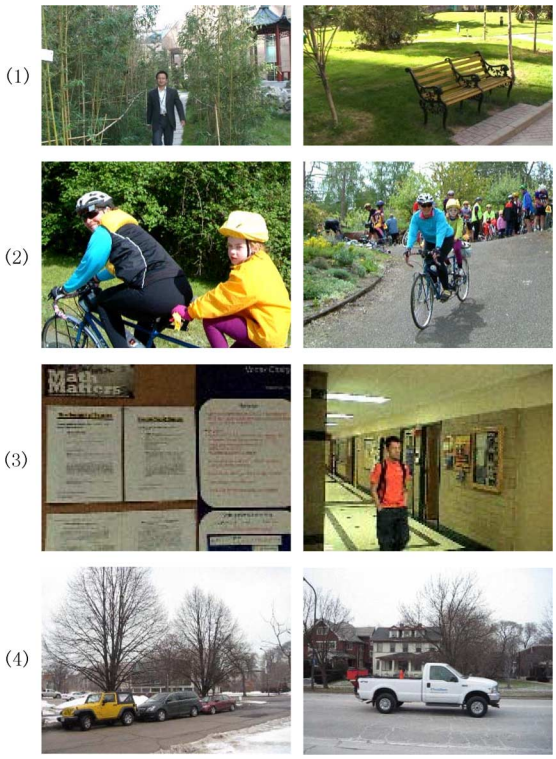


Fig. 6. Sample LR frames of the experiment data. The first column represents the group number of the frames. The original images are resized into the same size so that they can be seen more clearly.

TABLE I
Statistics of the Experiment Data

Group No.	1	2	3	4
No. of input sets	6	3	14	7
Capture device	Sony DCR-DVD905E	FujiFilm F601 Zoom	Sony DCR-TRV33	Canon A1100 IS
Frame rate	25 fps	15 fps	25 fps	30 fps
LR resolution	384×216	320×240	160×120	320×240
AWGN variance	0.02	0	0.2	0.05

TABLE II
Performances of Different Classifiers in PSNR (dB)

	In-group	Left-one	Cross-group
AdaboostM1 [36]	27.5256	27.6102	27.5762
Rforest10 [37]	28.5966	27.5476	27.3691
Rforest30 [37]	28.4280	27.7231	27.6309

TABLE III
AVERAGE BENCHMARK PERFORMANCE IN PSNR (dB)

Method	Benchmark performance
MAP [5]	27.0181
POCS [8]	22.5626
CMap [23]	25.5975
Edge [21]	26.7099
Random	25.7138
Best	28.5978
NI interpolation [11]	19.5240
Steering [33]	26.0764
Proposed (In-group)	28.4280
Proposed (Left-one)	27.7231
Proposed (Cross-group)	27.6309

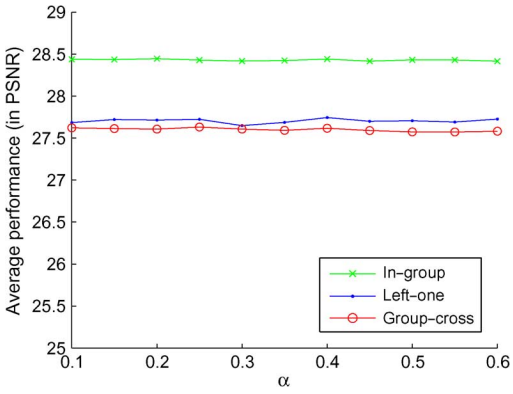


Fig. 7. Average benchmark performances in PSNR (in decibels) using different α [see (28)].

four-pixel wide. In this way, the proposed implementation classifies each well-registered block into four categories, which are suitable to be processed by MAP [5], POCS [8], confidence map [23], and the algorithm of [21], respectively.

B. Benchmark Scheme and its Results

The proposed benchmark scheme is based on a database of several input sets. Each input set contains the successive input LR frames for one single super-resolution task and the ground truth of the target HR image. For example, in our experiment, each input set contains seven successive LR frames and one original HR image of the fourth LR frame. We define notation s_i as the ith input set in the database.

The LR images in each s_i in the database may be of different image content, i.e., captured by different cameras and even of different sizes. The performance of a super-resolution algorithm varies under different conditions. Thus, given a database of sufficient input sets, the average performance of all the input sets is reasonable to evaluate the effectiveness of a super-resolution algorithm. Therefore, for a reconstruction-based super-resolution or image enhancement algorithm, we simply calculate the average performance (PSNR in our experiment) of all the input sets as the benchmark result of the algorithm.

We further divide the input database $\mathcal{S} = \{s_i\}$ into groups according to the characteristics of the input sets, where we use $\mathcal{G}_j$ to denote the jth group. In our experiment, input sets that are captured by the same capturing device are assigned to the same group; therefore, input sets in the same group have similar noise level and are of the same size. Three benchmark methods are introduced to evaluate the performance of a super-resolution algorithm. Each method assigns a different training set $\mathcal{T}(s_i)$ for each of the testing input set s_i:

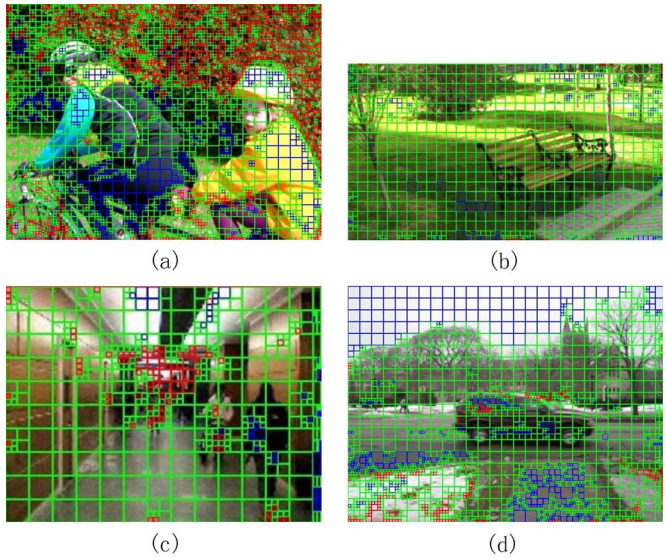


Fig. 8. First-level block division results of several images. Blue blocks represents flat ones, red for mismatched ones, and green for registered ones. The images are resized into approximately the same size to fit the page.

1) *In-group benchmark*: All other input sets in the same group as s_i are chosen as the following training samples:

$$\mathcal{T}_{\text{in-group}}(s_i) = \{s_k | s_k \in \mathcal{G}_j, i \neq k, \text{where } s_i \in \mathcal{G}_j\}. \quad (32)$$

In-group benchmark is the easiest of the three benchmarks, for the training set and the testing sample are in the same group and thus tend to be of very similar distribution.

2) *Left-one benchmark*: All other input sets in $\mathcal{S}$ are chosen as the following training samples:

$$\mathcal{T}_{\text{left-one}}(s_i) = \{s_k | s_k \in \mathcal{S}, i \neq k\}. \quad (33)$$

3) *Cross-group benchmark*: All inputs sets in $\mathcal{S}$ that are in different groups from s_i are chosen as the following training samples:

$$\mathcal{T}_{\text{cross-group}}(s_i) = \{s_k | s_k \notin \mathcal{G}_j, \text{where } s_i \in \mathcal{G}_j\}. \quad (34)$$

The cross-group benchmark usually fits the case of real super-resolution tasks where all the available training samples may be captured by different devices, of different image content, or of different noise level, and is obviously the toughest benchmark of the three.

The average performance of all the input sets s_i is calculated as the benchmark result for every benchmark method, which shows the performance under different situations.

In our experiment, four groups of input sets are used. The statistics of the input sets are shown in Table I. All videos are captured by handheld cameras and under various situations: indoor videos, outdoor videos, videos with translational global motion, videos with large local motion, etc. Some sample LR frames in the input sets are shown in Fig. 6.

Fig. 9. Image reconstruction results. Image (A) and (B) show the ground-truth HR images with the regions of interest marked in red boxes. (a)–(e) zoom the image details corresponding to the red boxes area. (a) LR image. (b) First-level block division results. Blue blocks represents flat ones, red for mismatched ones, and green for registered ones. (c) Results of the candidate algorithms. CMap, POCS, MAP, and Edge indicate the results of [5], [8], [21] and [23], respectively. (d) Results of the proposed algorithm. (e) HR ground truth.

We use three classification methods to test the performance of the proposed algorithm: boosting [36], random forest [37] with the number of trees 10 and with the number of trees 30, which are abbreviated as "adaboostM1", "rforest10" and "rforest30", respectively. The benchmark results of different classifiers are shown in Table II. Rforest30 performs the best on average. Thus, rforest30 is used as the classifier of the proposed algorithm. The idea of using the random forest classifier to select suitable candidate algorithms is also adopted in [39].

The comparative benchmark results of the proposed algorithm (using rforest30) are listed in Table III. The benchmark results in the "random" row in Table III are generated by randomly picking a candidate super-resolution algorithm for each block, whereas the results in the "best" row are generated by selecting the best candidate super-resolution algorithm for each block, both followed by the deblocking process. In addition to the candidate algorithms, we also show the benchmark results of two state-of-the-art super-resolution algorithms for comparison: the noniterative interpolation algorithm [11] and the steering kernel regression algorithm [33]. The software of which is provided by the authors of the corresponding papers. The algorithm proposed in [11] aims to minimize the frequency aliasing

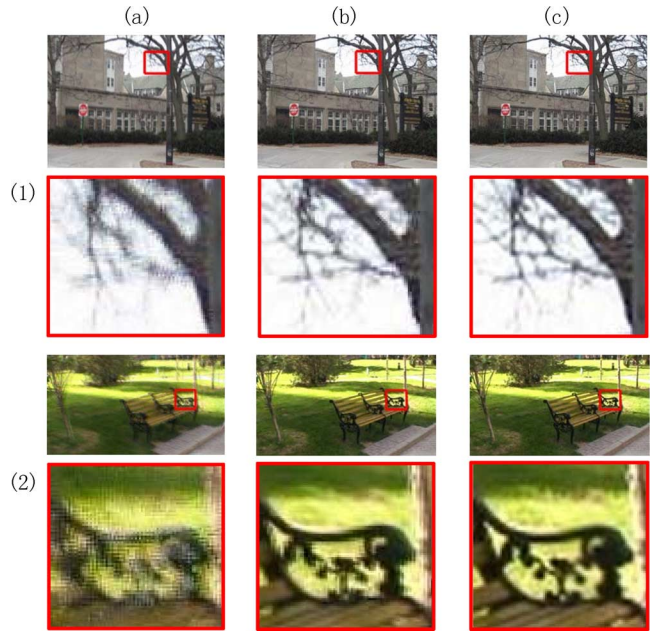


Fig. 10. Image reconstruction results. (a) Results of [11]. (b). Results of [33]. (c) Results of the proposed. (1) and (2) show the original and zoomed results of a super-resolution task, respectively.

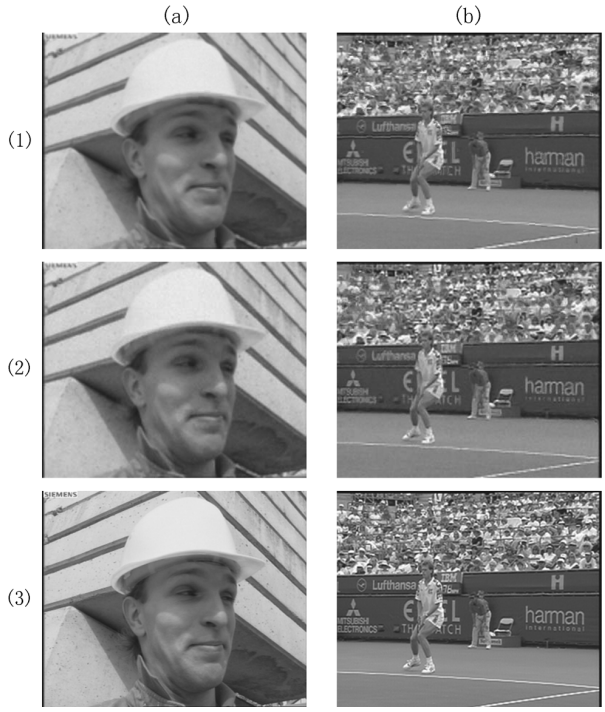


Fig. 11. Image reconstruction results on the data used in [33]. (1) Results of [33]. (2) Result of the proposed method. (3) Ground-truth image.

in super-resolution. The method [33] focuses on the problem of inaccurate motion estimation, which proposes an SR framework that avoid the process of explicit motion estimation. We can see that the proposed algorithm gets better benchmark results. The results of the last two benchmark methods (left-one and cross-group) are similar, which indicates that the proposed algorithm is robust to the statistics of the training set and extracts the essential characteristics of the candidate algorithms. Note that, although the method [11] produces good results when only affine motion is presented, it is not an algorithm robust to motion registration errors; therefore, it performs relatively worse under arbitrary motion fields.

The proposed algorithm is also robust to the parameters selected. The average performances adopting different α [see (28)] in the experiment are shown in Fig. 7. We can see the performance of the proposed algorithm is robust to the value of the parameter α since the average benchmark performances do not vary very much when the value of α changes. As is mentioned in Section VI-A, α is set as 0.25 in the experiment.

C. Image Reconstruction Results

Fig. 8 shows several first-level block division results of the images in the database. The bicycler and the child in Fig. 8(a) move quickly, and its difficult to accurately estimate motion fields within most parts of the trees on the background; therefore the blocks in the background are mostly classified into mismatched block category in the first-level. Blocks within part of the meadow in Fig. 8(b) and within the sky/road in Fig. 8(d) contain very little image content information, which are labeled as flat block category. Fig. 8(b) shows a static scene captured by a slow-moving camera, and most of the blocks within it are registered blocks. Only registered blocks are classified and processed in the second level using candidate super-resolution algorithms, whereas the flat ones and mismatched ones are processed directly according to Fig. 1.

In Fig. 9, some local synthesis results of the proposed algorithm compared with the candidate conventional algorithms are shown. We only show two representative results of the candidate algorithms for each block because some candidate results are similar to the proposed result and some candidate results are far worse in some specific cases. Sometimes, several candidate results can also be better than the synthesis result proposed as the proposed framework implementation does not always select the best candidate algorithm for each block. However, it prevents selecting bad candidate results most of the time and, on average, gets better performance (see Table III), which increases the algorithm robustness. In Fig. 10, the results of the proposed algorithm are compared with the results of [11] and [33]. The reconstruction results of [33] and the proposed method on the image data used in [33][1] are shown in Fig. 11. We can see that the proposed algorithm (adopting the second group in Table I as the training set) produces finer image details. See the word "harman" and the logo behind the player on the background of the images in (b). We also show some other HR image results of the proposed algorithm in Fig. 12 to get an overview. Note that all the HR images from the proposed algorithm shown in Figs. 9, 10, and 12 are the results of the cross-group benchmark.

The computational cost of the proposed algorithm depends on the area of different block categories. Only one candidate algorithm is applied within each block region. Image interpolation and denoising for flat blocks require very little computation compared with the other algorithms. Generally speaking, single-image enhancement for mismatched blocks is less time-consuming than candidate super-resolution algorithms. Thus, the proposed algorithm is more efficient than candidate super-resolution algorithms on average. However, this is not always the case because block classification costs additional computation,

[1]http://users.soe.ucsc.edu/~htakeda/SpaceTimeSKR.htm.

Fig. 12. Full HR image results of the proposed algorithm. (1) Input LR images. (2) Reconstruction result of the proposed algorithm.

and in the deblocking process, overlapping regions are generated where two algorithms need to be applied.

VII. Conclusion

A spatially adaptive block-based super-resolution framework is proposed in this paper. The target HR image plane is divided into adaptive-sized blocks in the framework. Each block can be assigned a suitable candidate algorithms. We also present an implementation of the framework to improve the super-resolution results under inaccurate motion registration and to make the best use of the advantages of different conventional super-resolution algorithms. One of the contributions of this paper is that we provide a way to combine a lot of related techniques together in the proposed framework, including rule-based and learning-based image/video classification, single-image enhancement, and image/video feature extraction.

In future work, self-adaptive algorithm parameters can be used also for blocks of the same category. We can also incorporate the framework into video compression and decompression algorithms to get decompressed video streams of higher quality.

Appendix A
Proof of the Statement in Section IV-B

The proposed deblocking process improves the image fidelity (in PSNR).

Proof: The PSNR (31) is a monotone decreasing function of the SSD, which is defined as

$$\mathrm{SSD}(H,I)=\sum_{x,y}(H(x,y)-I(x,y))^2 \tag{35}$$

where H is the original HR image and I is the estimated SR output.

Considering the overlapping area along the edge of two adjacent blocks using different algorithms, we assume that the blur kernel of the super-resolution result in one side of the overlapping area is G_1 and that the blur kernel of the other side is G_2; therefore, in the overlapping area, the SSD of the super-resolution result *without* deblocking is

$$SSD_1=\sum_{r<0}(H-G_1*H)^2+\sum_{r>0}(H-G_2*H)^2 \tag{36}$$

where the operator $*$ represents convolution and we omit notation (x,y) for simplicity. The SSD of the super-resolution result with deblocking process in the overlapping area is

$$SSD_2=\sum\left(H-(f^*(r)(G_1*H)+f^*(-r)(G_2*H))\right)^2. \tag{37}$$

Note that $f^*(r)+f^*(-r)=1$, and its reasonable to assume that G_1, G_2, and the statistics of H in the overlapping area are near space invariant since the overlapping area is usually small. Thus, we have

$$\begin{aligned}SSD_2 &\le \sum(f^*(r)(H-G_1*H))^2\\ &\quad+\sum(f^*(-r)(H-G_2*H))^2\\ &\doteq \frac{1}{2r_0}\int_{-r_0}^{r_0}(f^*(r))^2\,\mathrm{d}r\sum(H-G_1*H)^2\\ &\quad+\frac{1}{2r_0}\int_{-r_0}^{r_0}(f^*(-r))^2\,\mathrm{d}r\sum(H-G_2*H)^2\\ &\le \frac{1}{2}\sum(H-G_1*H)^2+\frac{1}{2}\sum(H-G_2*H)^2\\ &\doteq SSD_1.\end{aligned} \tag{38}$$

□

Acknowledgment

The authors would like to thank the support from the HP Labs China and the Computational Vision Group in Northwestern University.

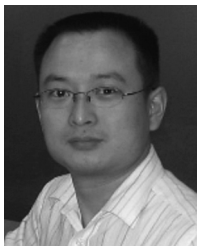
References

[1] R. Y. Tsai and T. S. Huang, "Multiframe image restoration and registration," *Adv. Comput. Vis. Image Process.*, vol. 1, pp. 317–339, 1984.
[2] S. C. Park, M. K. Park, and M. G. Kan, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Process. Mag.*, vol. 20, no. 3, pp. 21–36, May 2003.
[3] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion, and transparency," *J. Vis. Commun. Image Represent.*, vol. 4, no. 4, pp. 324–335, Dec. 1993.
[4] A. Tekalp, M. Ozkan, and M. Sezan, "High-resolution image reconstruction from lower-resolution image sequences and space-varying image restoration," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1992, pp. 169–172.

[5] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super-resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.
[6] G. Chantas, N. Galatsanos, and N. Woods, "Super-resolution based on fast registration and maximum a posteriori reconstruction," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1821–1830, Jul. 2007.
[7] S. Belekos, N. Galatsanos, and A. Katsaggelos, "Maximum *a posteriori* video super-resolution using a new multichannel image prior," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1451–1464, Jun. 2010.
[8] A. Patti, M. Sezan, and A. Murat Tekalp, "Super resolution video reconstruction with arbitrary sampling lattices and nonzero aperture time," *IEEE Trans. Image Process.*, vol. 6, no. 8, pp. 1064–1076, Aug. 1997.
[9] C. Fan, J. Zhu, J. Gong, and C. Kuang, "POCS super-resolution sequence image reconstruction based on improvement approach of Keren registration method," in *Proc. 6th Int. Conf. ISDA*, Oct. 16–18, 2006, vol. 2, pp. 333–337.
[10] H. Ji and C. Fermuller, "Robust wavelet-based super-resolution reconstruction: Theory and algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 649–660, Apr. 2009.
[11] A. Sanchez-Beato and G. Pajares, "Noniterative interpolation-based super-resolution minimizing aliasing in the reconstructed image," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1817–1826, Oct. 2008.
[12] W. Freeman and E. Pasztor, "Markov networks for super-resolution," in *Proc. 34th Annu. Conf. Inf. Sci. Syst.*, 2000.
[13] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proc. IEEE Comput. Soc. CVPR*, 2004, vol. 1, pp. I-275–I-282.
[14] W. Fan and D.-Y. Yeung, "Image hallucination using neighbor embedding over visual primitive manifolds," in *Proc. IEEE Conf. CVPR*, 2007, pp. 1–7.
[15] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. 12th Int. Conf. Comput. Vis.*, 2009, pp. 349–356.
[16] W. Freeman, T. Jones, and E. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar./Apr. 2002.
[17] M. Elad and D. Datsenko, "Example-based regularization deployed to super-resolution reconstruction of a single image," *Comput. J.*, vol. 52, no. 1, pp. 15–30, Jan. 2009.
[18] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.
[19] W. Zhao and H. Sawhney, "Is super-resolution with optical flow feasible?," in *Proc. 7th Eur. Conf. Comput. Vis.—Part I*, 2002, pp. 599–613.
[20] S. Dai, M. Han, W. Xu, Y. Wu, and Y. Gong, "Soft edge smoothness prior for alpha channel super resolution," in *Proc. IEEE Conf. CVPR*, Jun. 17–22, 2007, pp. 1–8.
[21] Y. Cha and S. Kim, "Edge-forming methods for color image zooming," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2315–2323, Aug. 2006.
[22] J. Kim, R. Park, and S. Yang, "Super-resolution using POCS-based reconstruction with artifact reduction constraints," *Vis. Commun. Image Process.*, vol. 5960, p. 596 05B, 2005.
[23] M. Chen, "Dynamic content adaptive super-resolution," *Image Anal. Recognit.*, vol. 3212, Lecture Notes in Computer Science, pp. 220–227, 2004.
[24] A. Yau, N. Bose, and M. Ng, "An efficient algorithm for superresolution in medium field imaging," *Multidimensional Syst. Signal Process.*, vol. 18, no. 2, pp. 173–188, Sep. 2007.
[25] Y. He, K.-H. Yap, L. Chen, and L.-P. Chau, "A nonlinear least square technique for simultaneous image registration and super-resolution," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2830–2841, Nov 2007.
[26] L. Baboulaz and P. L. Dragotti, "Exact feature extraction using finite rate of innovation principles with an application to image super-resolution," *IEEE Trans. Image Process.*, vol. 18, no. 2, pp. 281–298, Feb. 2009.
[27] H. Shen, L. Zhang, B. Huang, and P. Li, "A map approach for joint motion estimation, segmentation, and super resolution," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 479–490, Feb. 2007.
[28] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint map registration and high-resolution image estimation using a sequence of undersampled images," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1621–1633, Dec. 1997.
[29] A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian, "Image and video super-resolution via spatially adaptive block-matching filtering," in *Proc. Int. Workshop LNLA*, 2008.
[30] M. Protter and M. Elad, "Super resolution with probabilistic motion estimation," *IEEE Trans. Image Process.*, vol. 18, no. 8, pp. 1899–1904, Aug. 2009.
[31] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the nonlocal-means to super-resolution reconstruction," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 36–51, Jan. 2009.
[32] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 349–366, Feb. 2007.
[33] H. Takeda, P. Milanfar, M. Protter, and M. Elad, "Super-resolution without explicit subpixel motion estimation," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 1958–1975, Sep. 2009.
[34] H. Su, L. Tang, D. Tretter, and J. Zhou, "A practical and adaptive framework for super-resolution," in *Proc. 15th IEEE ICIP*, 2008, pp. 1236–1239.
[35] D. Marpe, T. Wiegand, and G. Sullivan, "The H.264/MPEG4 advanced video coding standard and its applications," *IEEE Commun. Mag.*, vol. 44, no. 8, pp. 134–143, Aug. 2006.
[36] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. Int. Conf. Mach. Learn.*, 1996, pp. 148–156.
[37] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
[38] M. Black and P. Anandan, "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Comput. Vis. Image Understanding*, vol. 63, no. 1, pp. 75–104, 1996.
[39] O. Mac Aodha, G. Brostow, and M. Pollefeys, "Segmenting video into classes of algorithm-suitability," in *Proc. IEEE Conf. CVPR*, 2010, pp. 1054–1061.

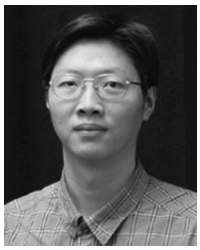
Heng Su received the B.S. degree (with hons.) from Tsinghua University, Beijing, China, in 2006, and is currently working toward the Ph.D. degree with the Department of Automation, Tsinghua University.

His research interests include super-resolution image reconstruction, computer vision, and image and video processing.



Liang Tang received the Ph.D. degree in information and communication engineering from Xidian University, Xi'an, China, in 2004.

From 2004 to early 2007, he was a Post-Doctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China. After that, he worked for HP Labs China as a Research Scientist on image processing and computer vision from 2007 to 2010. Currently, he is leading the R&D efforts on computer vision algorithms and applications on mechatronics equipments in the No. 45 Research Institute of the China Electronics Technology Group Corporation, Beijing, China. His research interests include computer vision, image processing, pattern recognition and their applications on mechatronics equipments.



Ying Wu (SM'06) received the B.S. degree from Huazhong University of Science and Technology, Wuhan, China, in 1994; the M.S. degree from Tsinghua University, Beijing, China, in 1997; and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC), Urbana, in 2001.

From 1997 to 2001, he was a Research Assistant with the Beckman Institute for Advanced Science and Technology, UIUC. During summer 1999 and 2000, he was a Research Intern with Microsoft Research, Redmond, WA. In 2001, he joined the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL, as an Assistant Professor. He is currently an Associate Professor of electrical engineering and computer science with Northwestern University. His current research interests include computer vision, image and video analysis, pattern recognition, machine learning, multimedia data mining, and human–computer interaction.

Dr. Wu serves as an Associate Editor for IEEE TRANSACTIONS ON IMAGE PROCESSING, the SPIE *Journal of Electronic Imaging*, and the IAPR *Journal of Machine Vision and Applications*. He was the recipient of the Robert T. Chien Award at UIUC in 2001 and the National Science Foundation CAREER Award in 2003.

Daniel Tretter received the B.S. degree in mathematics and electrical engineering from the Rose–Hulman Institute of Technology, Terre Haute, IN, and the M.S.E.E. and Ph.D. degrees from Purdue University, West Lafayette, IN.

He is a Research Manager with the Printing and Content Delivery Laboratory, HP Labs Palo Alto, Hewlett–Packard Company, Palo Alto, CA. His project team develops technologies that leverage multimedia analysis for improved media management and creation of custom media presentations. He is the author of a number of technical papers and two book chapters. He is the coauthor of 55 patents in the area of image processing, with approximately 12 more pending. His current research interests include the management, browsing, and presentation of personal media collections.

Jie Zhou (M'01–SM'04) received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995.

From 1995 to 1997, he was a postdoctoral fellow with the Department of Automation, Tsinghua University, Beijing, China. Currently, he is a Full Professor with the Department of Automation, Tsinghua University. He is the author of more than 100 papers in peer-reviewed international journals and conferences. His research interests include pattern recognition, computer vision, and data mining.

Dr. Zhou serves as an Associate editor for the *International Journal of Robotics and Automation* and *Acta Automatica Sinica*. He was the recipient of the Best Doctoral Thesis Award from HUST in 1995, the First Class Science and Technology Progress Award from the Ministry of Education (MOE) in 1998, the Excellent Teaching Award from Tsinghua University in 2003, and the Best Advisor Award from Tsinghua University in 2004 and 2005. He was selected as one of the outstanding scholars of MOE in 2005.