

Research Article

A Novel Adaptive Directional Interpolation Algorithm for Digital Video Resolution Enhancement

Dong Sun ¹, Qingqing Xie ¹, Teng Li¹, Yixiang Lu ¹, De Zhu ², and Qingwei Gao ¹

¹College of Electrical Engineering and Automation, Anhui University, Hefei, Anhui, China 230601

²Network and Information Center, Anhui University, Hefei, Anhui, China 230601

Correspondence should be addressed to Qingwei Gao; qingweigao@ahu.edu.cn

Received 25 June 2020; Accepted 23 November 2020; Published 16 December 2020

Academic Editor: Huimin Lu

Copyright © 2020 Dong Sun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, a novel digital video resolution enhancement algorithm based on adaptive directional interpolation is proposed, where the directionality of the edge structure and the nonlocal self-similarity prior within the current frame as well as its adjacent frames are both considered. First, we establish the regularization equation that conforms to the prior model of a video frame and then take the classic bicubic interpolation result as the initial estimation to iteratively solve the restoration equation, in which the edge structures and contours in low resolution (LR) input are reconstructed to estimate and refine the desired high resolution (HR) output. Experimental results show that the proposed algorithm can effectively enhance the clarity of a video frame, with satisfying subjective visual quality and PSNR value.

1. Introduction

Videos and images are the main sources of information for humans. According to statistics, more than 80% of the information we receive from the outside world comes from vision. With the development of digital mobile communication and computer technology, various novel applications such as distance education, video on demand, telemedicine, and multi-person online video conference have appeared, promoting the revolution of productivity and social progress. In the meantime, the image quality of digital video has also been desired higher and higher, where the clarity index comes from standard definition to high definition (HD) and ultra-high definition, as well as the corresponding resolution index also comes from 480p to 720p, 1080p, and 2160p (4K). On the one hand, these improvements in clarity and resolution can meet the increasing demand of end users and provide better image quality; on the other hand, while high-resolution video provides more details in content, it also adds burdens to the entire production and consumption ecosystem: more expensive capture and storage devices on the

image acquisition side, additional computing resource requirement for video editing on the media creation side, and more data transmission pressure on the communication network side. All these above have become important factors that restrict further improvement of video clarity and quality. In order to solve this problem, a common way is to use an image postprocessing procedure where the LR input frame is interpolated by a superresolution method [1–7], leading to a resolution-enhanced HR one. This software-based technique does not change the existing image acquisition and data transmission systems and thus is of great value in fields of videotelephony, virtual reality, augmented reality, and HD video games.

Natural images are highly structured, which reflects the strong time-spatial redundancy and self-similarity underlying pixels and performs a key role in solving inverse problems such as image denoising, deblurring, inpainting, and superresolution. By considering the fact that the human visual system is sensitive to the image edge structure [7–11], a novel digital video resolution enhancement algorithm via adaptive directional filtering is proposed in

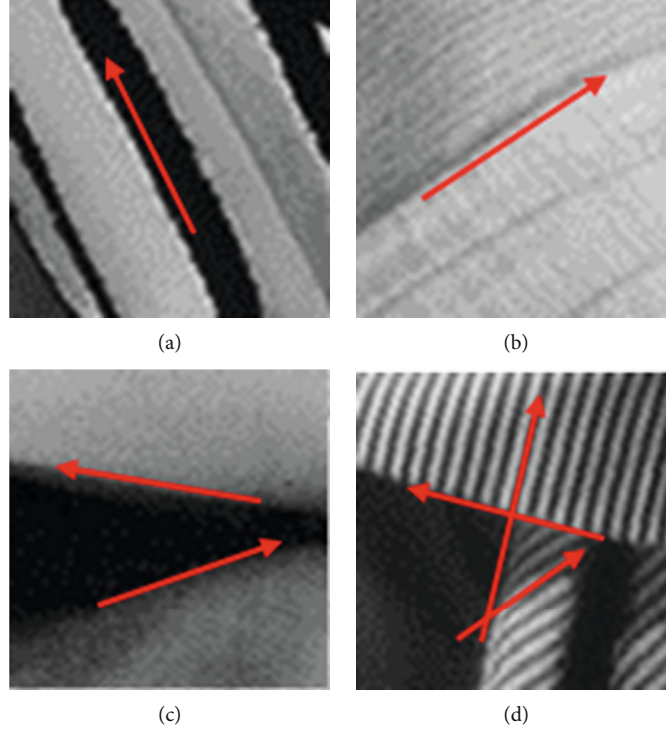


FIGURE 1: Directional regularity in natural images: (a) an image patch contains one main direction (120 degrees); (b) an image patch contains one main direction (40 degrees); (c) an image patch contains two main directions (30 degrees and 170 degrees); (d) an image patch contains three main directions (40 degrees, 80 degrees, and 165 degrees).

this paper, in which the characteristics of the edge contour and the nonlocal self-similarity within current frame as well as the corresponding adjacent frames are both considered. We first establish the regularization equation that conforms to the prior model of a video frame and then take the classic bicubic interpolation result as the initial estimation to iteratively solve the restoration equation, where the edge structures and contours in LR input are reconstructed to estimate and refine the desired HR output.

The rest of the sections are organized as follows. In Section 2, we introduce the core idea of the proposed adaptive directional interpolation scheme for estimating the missing details of the LR image and then use the nonlocal self-similarity prior to further improve the interpolation performance. The details of the video resolution enhancement algorithm are provided in Section 3. Section 4 presents the experimental validations of the proposed algorithm and comparison with the classic bicubic interpolation method; conclusions are drawn in Section 5.

2. The Core Idea

Directional regularity has widely existed in textures, edges, and contours of natural images (shown as in Figure 1). Denote vector $f_i \in \mathbb{R}^{n^2}$ as the image patch centered around the i th pixel and with sizes $n \times n$, and $L_\theta \in \mathbb{R}^{n^2 \times n^2}$ as the filter matrix corresponding to the directional filter with angle θ (in this paper, the directional controllable steerable filter [12] is

used). Obviously, the filtered vector $L_\theta f_i$ is the sparsest (namely, $L_\theta f_i$ is approximate to zero) when θ is parallel with the main direction of f_i . Generally, an image patch may include more than one main direction due to its complexity (examples are shown in Figures 1(c) and 1(d)); we can search for these direction angles using the following algorithm:

In our previous works [3, 13], we have shown the details to construct a blurring matrix from its corresponding linear degradation operator (as well as the downsampling matrix \mathbf{H}). Here, we simply present the steps to construct the directional filter matrix \mathbf{L} from a 2-D filter kernel \mathbf{B} , as follows:

- (i) Let \mathbf{L} be a $n^2 \times n^2$ zero matrix;
- (ii) For each pixel of the filtered image patch $\mathbf{d} = \mathbf{L}f_i$:
 - (a) Compute the 2-D coordinate (r, c) of pixel $d[i]$ from its 1-D index i ;
 - (b) For each element $B[u][v]$ of filter kernel \mathbf{B} , set the element $L[i][(c-v-1)n^2+r-u] = B[u][v]$.

The structure of filter matrix \mathbf{L} is presented in Figure 2.

Figure 3 shows the main direction searching results of test images *barbara* and *butterfly* using the algorithm above.

Denote $\mathbf{y}_i = \mathbf{H}f_i$ as the LR image patch, where $\mathbf{H} \in \mathbb{R}^{m^2 \times n^2}$ is the downsampling matrix [3]. When the downsampling factor D is an integer, we have $m = n/D$, and the corresponding LR input can be represented as $y(h, v) = f(h/D, v/D)$. With the constraint of the directional regularity posed above,

Main direction searching

Partition \mathbf{f} into overlapping patches $\{f_1, f_2, \dots\}$, and for each patch, do the following steps:

- **Initialization:** Set main direction angle set $\mathbf{S} = \emptyset$, candidate angle set $\Theta = \{\theta_1, \theta_2, \dots, \theta_K\}$, the largest number of direction angles P . Set start point $\mathbf{d} = \mathbf{f}_i$.
- **Main loop** (repeat P times):
 - Calculate the filtering result $\mathbf{L}_{\theta_1} \mathbf{d}, \mathbf{L}_{\theta_2} \mathbf{d}, \dots, \mathbf{L}_{\theta_K} \mathbf{d}$;
 - Find the best angle $\theta_{opt} = \operatorname{argmin}_{\theta_j} \|\mathbf{L}_{\theta_j} \mathbf{d}\|_1$;
 - Update $\mathbf{S} \leftarrow \mathbf{S} \cup \{\theta_{opt}\}$, $\Theta \leftarrow \Theta \setminus \{\theta_{opt}\}$ and $\mathbf{d} \leftarrow \mathbf{L}_{\theta_{opt}} \mathbf{d}$ for the next iteration.
- **Output:** The main direction angle set \mathbf{S} of the i th image patch \mathbf{f}_i .

ALGORITHM 1:

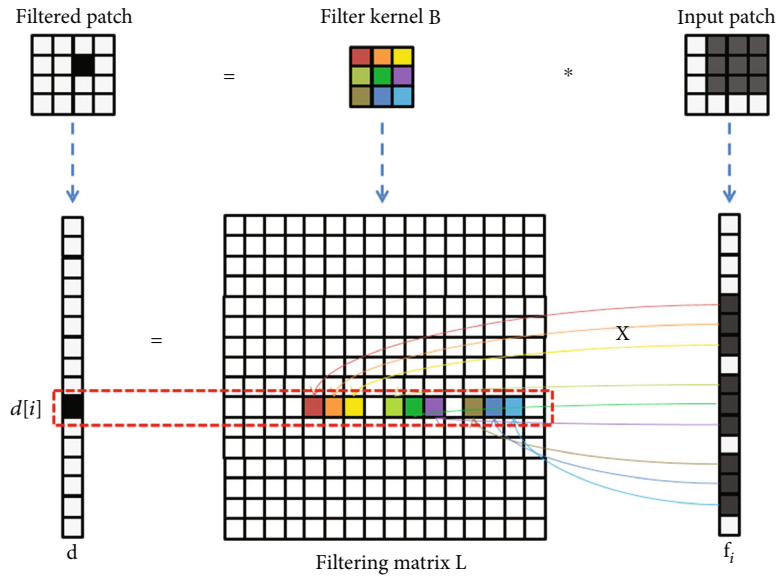


FIGURE 2: Structure of the directional filter matrix.

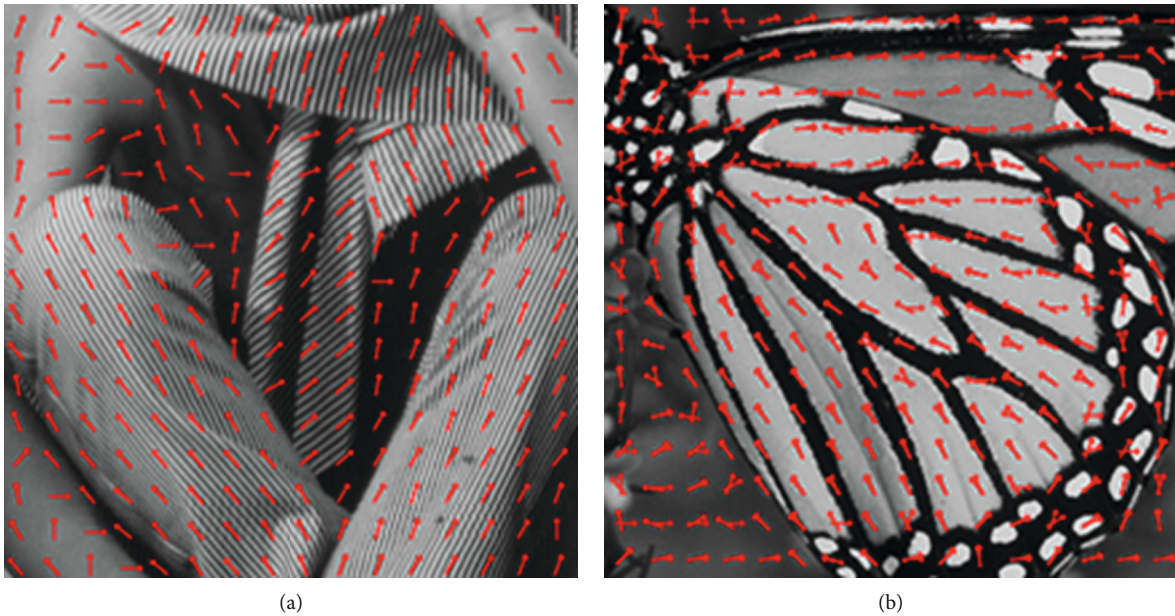


FIGURE 3: Main direction searching results: (a) *barbara* ($P = 1$); (b) *butterfly* ($P = 2$). Directions with similar degrees are merged.

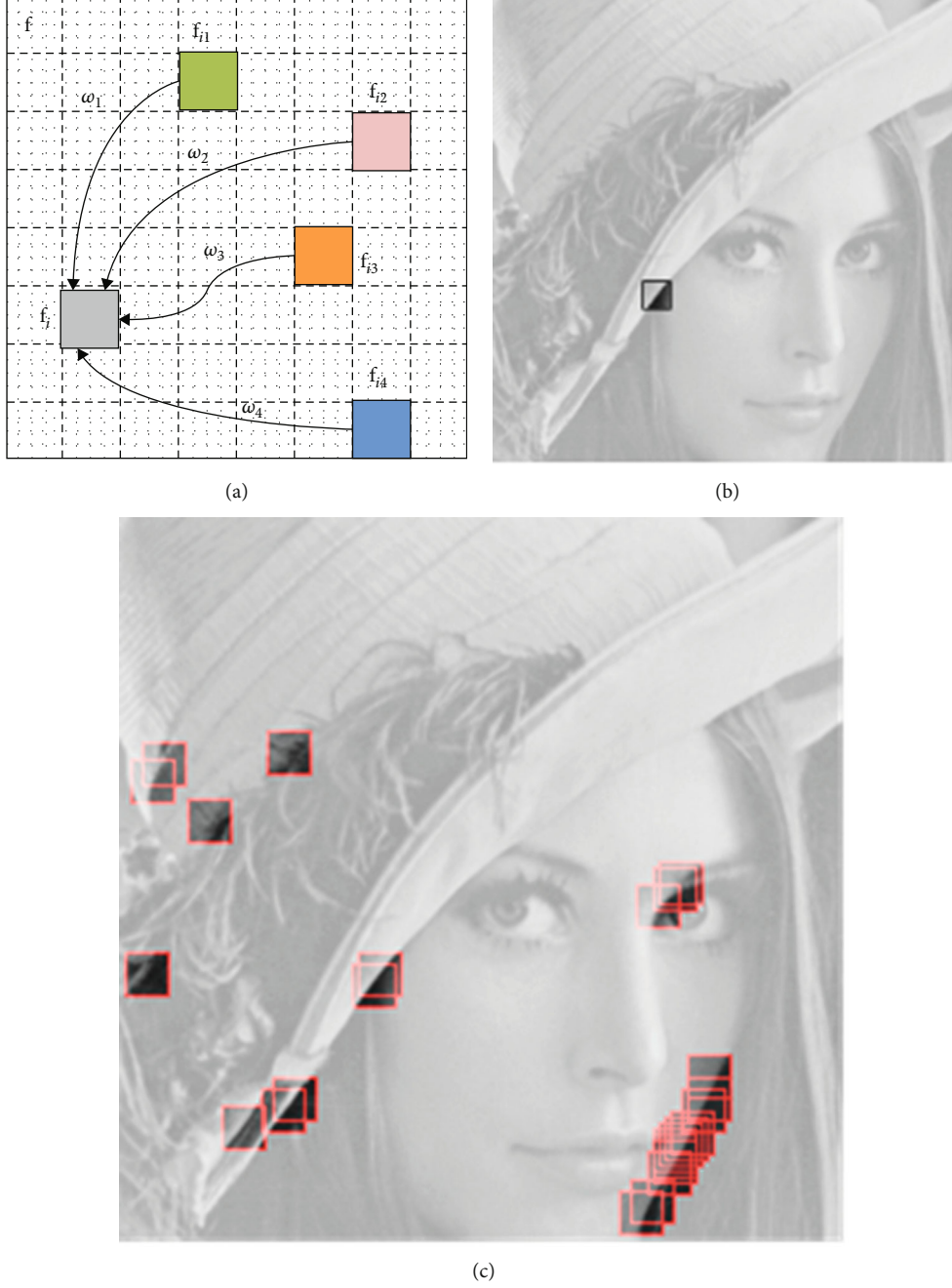


FIGURE 4: The NAR image model. As an example of (a), one image patch in (b) can be linearly represented by several nonlocal neighbors in (c).

the following interpolation equation can be used to estimate the original HR patch \mathbf{f}_i that

$$\mathbf{f}_i = \operatorname{argmin}_{\mathbf{f}_i} \|\mathbf{y}_i - \mathbf{H}\mathbf{f}_i\|_2^2 + \lambda \|\mathbf{L}_i \mathbf{f}_i\|_2^2, \quad (1)$$

where λ is the regularization parameter and $\mathbf{L}_i = \prod_{p=1}^P \mathbf{L}_{\theta_p}$ is the adaptive directional filter matrix. This equation posed above has the well-known closed-form solution

$$\mathbf{f}_i = (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{L}_i^T \mathbf{L}_i)^{-1} \mathbf{H}^T \mathbf{y}_i. \quad (2)$$

It is easy to know from the structure of the downsampling matrix \mathbf{H} that $\mathbf{H}^T \mathbf{H}$ is diagonal. For the downsampling factor $D = 2$, we have

$$\mathbf{H}^T \mathbf{H} = \begin{bmatrix} \mathbf{E}_n & & & & \\ & \mathbf{0}_n & & & \\ & & \dots & & \\ & & & \mathbf{E}_n & \\ & & & & \mathbf{0}_n \end{bmatrix} \in \mathbb{R}^{n^2} \times n^2, \quad (3)$$

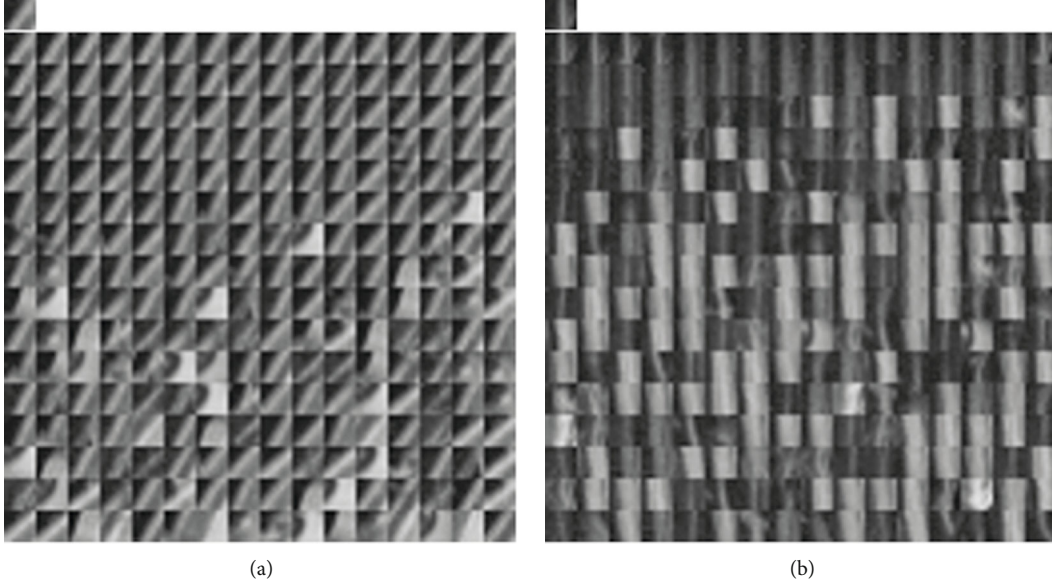


FIGURE 5: Image patch (top-left, $n = 8 \times 8$) and its adaptive dictionary (bottom, $M = 256$).

where $\mathbf{E}_n = \text{diag}(1, 0, 1, 0, \dots, 1, 0) \in \mathbb{R}^{n \times n}$ and $\mathbf{0}_n \in \mathbb{R}^{n \times n}$ is a zero matrix. Plugging the SVD decomposition $\mathbf{L}_i^T \mathbf{L}_i = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T = \mathbf{U} \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{n^2}) \mathbf{U}^T$ into expression (2), this leads to

$$\begin{aligned} \hat{\mathbf{f}}_i &= \mathbf{U}(\mathbf{H}^T \mathbf{H} + \mathbf{\Lambda})^{-1} \mathbf{U}^T \mathbf{H}^T \mathbf{y}_i \\ &= \begin{bmatrix} 1 + \theta_1 & & & & & & & & \\ & \theta_2 & & & & & & & \\ & & 1 + \theta_3 & & & & & & \\ & & & \dots & & & & & \\ & & & & & & & & \theta_{n^2} \end{bmatrix}^{-1} \mathbf{U}^T \mathbf{H}^T \mathbf{y}_i. \end{aligned} \quad (4)$$

Recall that $\mathbf{L}_i \mathbf{f}_i \approx \mathbf{0}$, and thus, \mathbf{L}_i is approximately singular, implying that one or more singular values of $\mathbf{L}_i^T \mathbf{L}_i$ are close to zero, and therefore, the inverse of the restoration kernel $\mathbf{H}^T \mathbf{H} + \lambda \mathbf{L}_i^T \mathbf{L}_i$ is ill-posed that can not be well handled. To solve this problem, we explore the self-similarity prior widely existing in natural images to further improve the interpolation performance. In this paper, the nonlocal autoregressive (NAR) model of images [14] is used to add additional constraint to the restoration kernel and reduce the degree of freedom of desired unknown pixels; this will help to yield a more stable result.

According to our previous works [15–17], we show that each patch in an image can be approximately represented as a linear combination of M nonlocal neighbors at different locations (shown as in Figure 4) that

$$\mathbf{f}_i \approx \sum_{j=1}^M \omega_j \mathbf{f}_{ij} = \mathbf{F}_i \boldsymbol{\omega}_i. \quad (5)$$

The neighbor set $\mathbf{F}_i = [\mathbf{f}_{i1}, \mathbf{f}_{i2}, \dots, \mathbf{f}_{iM}] \in \mathbb{R}^{n \times M}$ consists of M nonlocal patches around \mathbf{f}_i , which can be seen as an adaptive local dictionary that refers to the target vector \mathbf{f}_i , and the corresponding representation coefficient ω_i can be easily computed by ridge regression

$$\boldsymbol{\omega}_i = (\mathbf{F}_i^T \mathbf{F}_i + \gamma n \mathbf{I})^{-1} \mathbf{F}_i^T \mathbf{f}_i, \quad (6)$$

where the parameter γ is set manually to lead to the best results. Moreover, we have also proved in [15, 17] that ω_i is sparse when the atoms of \mathbf{F}_i are similar to \mathbf{f}_i in terms of normalized inner products. Considering that sparsity is very powerful that is broadly used in solving various inverse problems and has shown the ability to handle the image superresolution task [3, 6, 14, 15, 18], we here propose the following algorithm (Algorithm 2) to construct the adaptive dictionary \mathbf{F}_i :

Figure 5 shows the dictionary construction results of two patches of test images *lena* using the algorithm above. For video sequence, the above algorithm is also adapted to construct a dictionary for image patch of frames. At this time, each atom of \mathbf{F}_i comes from those nonlocal neighbors belonging to the current frame and its adjacent Q frames, shown as in Figure 6. Considering that video scene changes smoothly for most time, the differences between neighbor frames are small; this means it will be easier to find more similar candidate patches and thus finally leads to a sparser/better representation coefficient ω_i , which helps in improving the interpolation performance further.

Replacing the constraint posed in (5) by an equivalent penalty and adding it to Equation (1), we obtain

$$\hat{\mathbf{f}}_i = \arg \min_{\mathbf{f}_i} \|\mathbf{y}_i - \mathbf{H} \mathbf{f}_i\|_2^2 + \lambda \|\mathbf{L}_i \mathbf{f}_i\|_2^2 + \mu \|\mathbf{f}_i - \mathbf{F}_i \boldsymbol{\omega}_i\|_2^2. \quad (7)$$

Combining this equation with Equation (6), we get the desired HR patch estimator

Adaptive dictionary construction

Partition \mathbf{f} into overlapping patches $\{\mathbf{f}_1, \mathbf{f}_2, \dots\}$, and for each patch, do the following steps:

- **Initialization:** Set nonlocal neighbor number M and search window size W .
- **Dictionary construction:**
 - Sweep over all possible patches $\mathbf{f}_{i1}, \mathbf{f}_{i2}, \dots$ over the searching window centered around \mathbf{f}_i , and compute the normalized candidate atom set $\mathbf{G}_i = [(\mathbf{f}_{i1}/\|\mathbf{f}_{i1}\|), (\mathbf{f}_{i2}/\|\mathbf{f}_{i2}\|), \dots]$;
 - Compute the normalized inner product vector $\mathbf{r} = \mathbf{G}_i^T \mathbf{f}_i$;
 - Select the atoms with the largest M values in $|\mathbf{r}|$ to construct dictionary \mathbf{F}_i .
- **Output:** The adaptive dictionary \mathbf{F}_i of the i th image patch \mathbf{f}_i .

ALGORITHM 2:

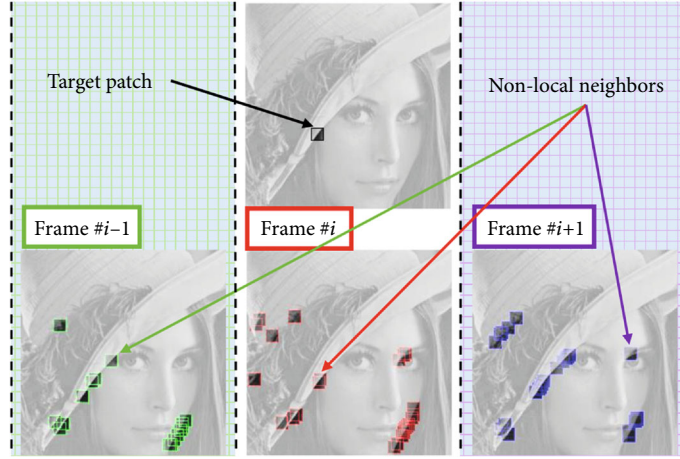


FIGURE 6: Dictionary construction. As an illustration, for a target patch in the i th frame of a video sequence, the corresponding adaptive dictionary is composed of nonlocal neighbors scattered over frames i , $i-1$, and $i+1$ (take $Q=1$ for example).

Resolution enhancement algorithm

For each LR frame \mathbf{y} of the input digital video sequence, do the following steps:

- **Initialization:** Set $\tilde{\mathbf{f}}$ the bicubic interpolation of \mathbf{y} .
- **Main loop** (repeat C times):
 - Use Algorithm 1 to search the main direction for each patch of $\tilde{\mathbf{f}}$, calculate the corresponding adaptive directional filter matrix \mathbf{L}_i ;
 - Use Algorithm 2 to construct the adaptive dictionary \mathbf{F}_i ;
 - Taking $\tilde{\mathbf{f}}$ as an initial estimation of the desired HR output \mathbf{f}_i , use Equation (8) to compute the resolution enhancement result \mathbf{f}_i ;
 - Update $\tilde{\mathbf{f}} \leftarrow \mathbf{f}_i$ for the next iteration when all image patches have been restored.
- **Output:** The resolution enhanced output \mathbf{f} .

ALGORITHM 3:

$$\mathbf{f}_i = (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{L}_i^T \mathbf{L}_i + \mu \mathbf{I})^{-1} \left(\gamma \mathbf{F}_i (\mathbf{F}_i^T \mathbf{F}_i + \gamma n \mathbf{I})^{-1} \mathbf{F}_i^T \tilde{\mathbf{f}}_i + \mathbf{H}^T \mathbf{y}_i \right). \quad (8)$$

Contrast the expression above with formula (2), we can see that the restoration kernel is full rank now, while keeping the advantage of diagonal, leading to a cheap computation of matrix inversion.

3. Video Resolution Enhancement Algorithm

To sum up, we use the interpolation algorithm (Algorithm 3) listed below for digital video resolution enhancement:

A graphic demonstration of this algorithm is displayed in Figure 7.

In each interpolation loop, the time consumption $T_{\text{loop}}(N)$ mainly consists of three parts, including the main direction searching $T_m(N)$, the adaptive dictionary constructing

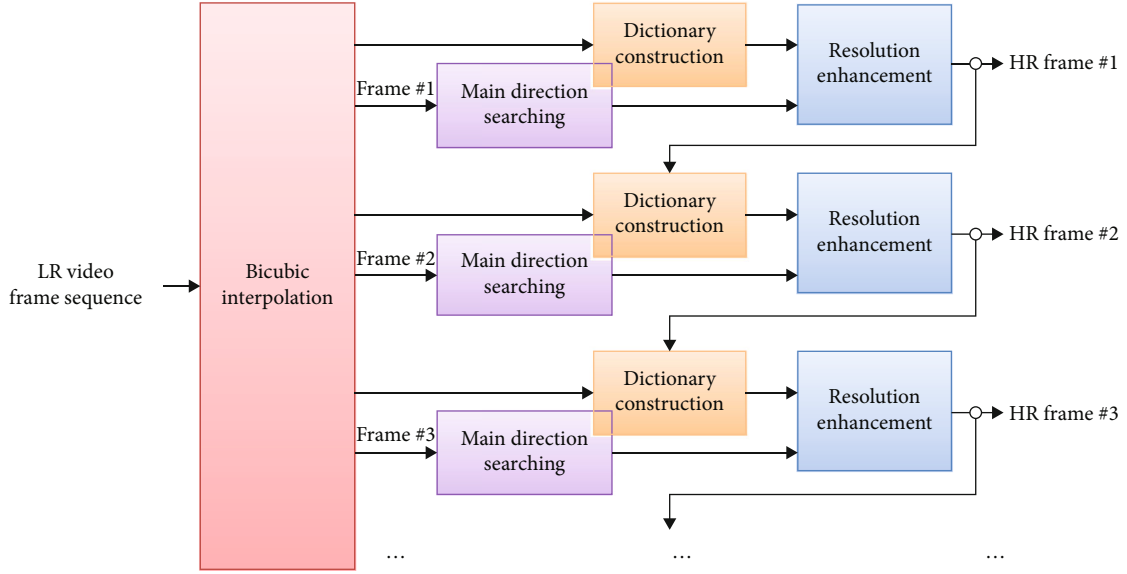


FIGURE 7: The flowchart of the proposed resolution enhancement algorithm for a video sequence.

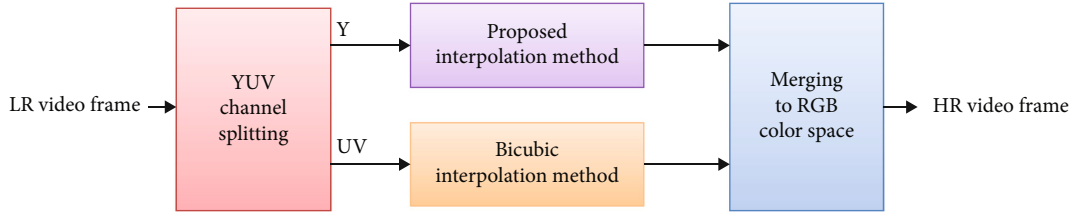


FIGURE 8: The flowchart of the proposed resolution enhancement algorithm for a color video sequence.

$T_a(N)$, and the HR output estimating $T_e(N)$, where N denotes the size of the LR input frame. That is

$$T_{\text{loop}}(N) = T_m(N) + T_a(N) + T_e(N). \quad (9)$$

For the first term $T_m(N)$, we know from Algorithm 1 that searching each direction for every target patch needs K filtering operations. Considering the fact that filtering a fixed-size image patch with size $n \times n$ can surely be done in constant time t_1 , therefore

$$T_m(N) = (N - n + 1)^2 \cdot (P \cdot K \cdot t_1) \sim O(N^2). \quad (10)$$

For the second term $T_a(N)$, we need to sweep over W^2 candidate patches around each target LR patch for searching atoms. Similarly, since the normalization and inner product computing can also be finished in constant time t_2 , thus

$$T_a(N) = (N - n + 1)^2 \cdot (W^2 \cdot t_2 + T_{\text{top}}(M)) \sim O(N^2) + O(N^2) \cdot T_{\text{top}}(M). \quad (11)$$

In the above expression, $T_{\text{top}}(M)$ represents the time consumption of selecting the top M largest elements from vector $|\mathbf{r}| = |\mathbf{G}_i^T \mathbf{f}_i| \in \mathbb{R}^{W^2}$, where this task can be simply implemented by a fast ordering algorithm with time com-

plexity $O(W^2 \log(W))$, and this leads to

$$T_a(N) \sim O(N^2) + O(N^2) \cdot O(W^2 \log(W)) \sim O(N^2). \quad (12)$$

For the last term $T_e(N)$, the time consumption is mainly determined by the computation of the inverse matrices $(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{L}_i^T \mathbf{L}_i + \mu \mathbf{I})^{-1}$ and $(\mathbf{F}_i^T \mathbf{F}_i + \gamma n \mathbf{I})^{-1}$. For the reason that the size of \mathbf{H} , \mathbf{L}_i , \mathbf{F}_i , and \mathbf{I} are fixed and indifferent to N , thus these operations can also be done in constant time t_3 . We have

$$T_e(N) = (N - n + 1)^2 \cdot t_3 \sim O(N^2). \quad (13)$$

Plugging Equations (10), (12), and (13) into (9), we obtain

$$T_{\text{loop}}(N) \sim O(N^2) + O(N^2) + O(N^2) \sim O(N^2). \quad (14)$$

The equation above means that the computational complexity of our proposed interpolation algorithm is proportional to the pixel number (N^2) of the LR input frame.

For color video sequence interpolation, the YUV color model can be considered: we start by splitting the input color frame into luminance channel and chrominance channel and then enhance each channel using the proposed algorithm and classic bicubic interpolation, respectively. The final

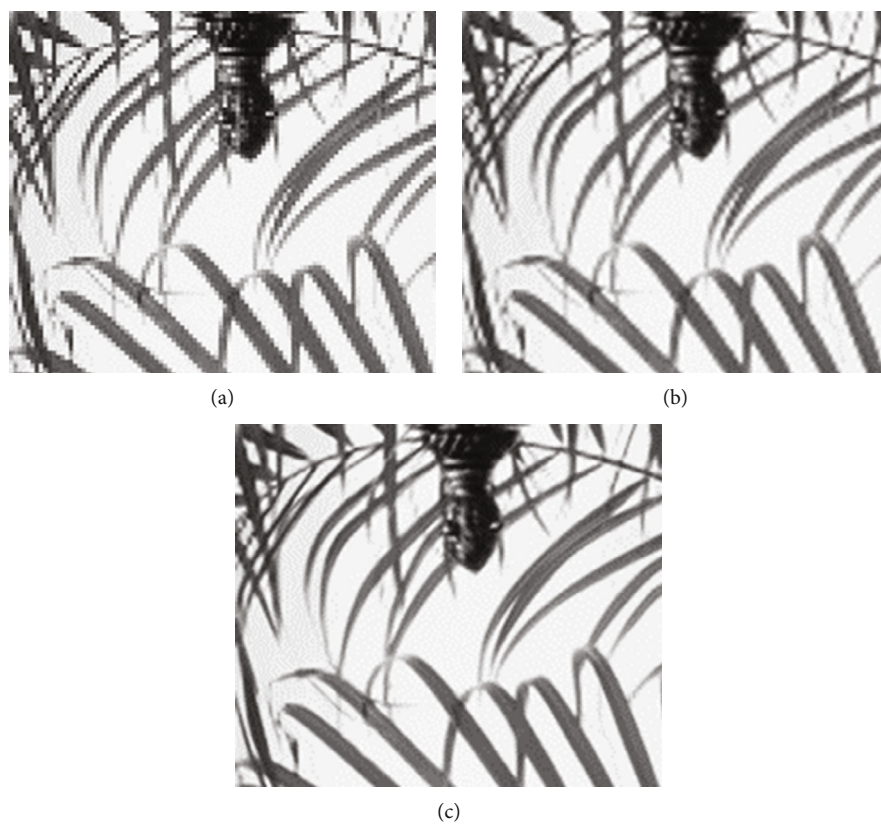


FIGURE 9: LR image *leaves* and the 2×2 interpolation results: (a) LR image; (b) bicubic (PSNR = 26.64); (c) proposed (PSNR = 29.23).

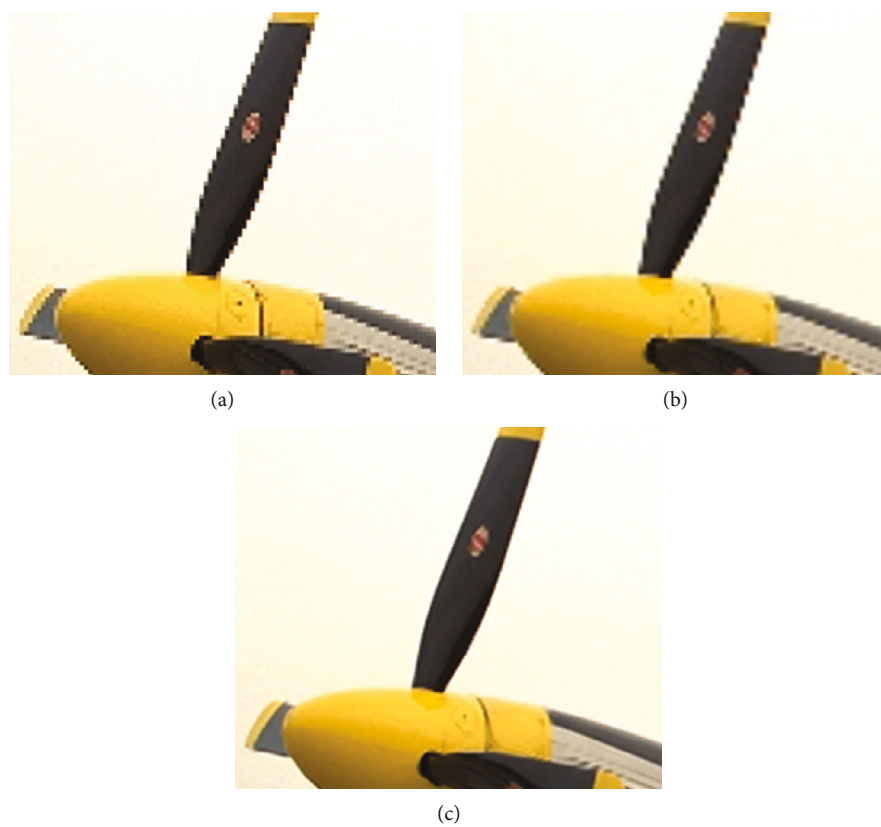


FIGURE 10: LR image *plane* and the 2×2 interpolation results: (a) LR image; (b) bicubic (PSNR = 29.59); (c) proposed (PSNR = 32.78).

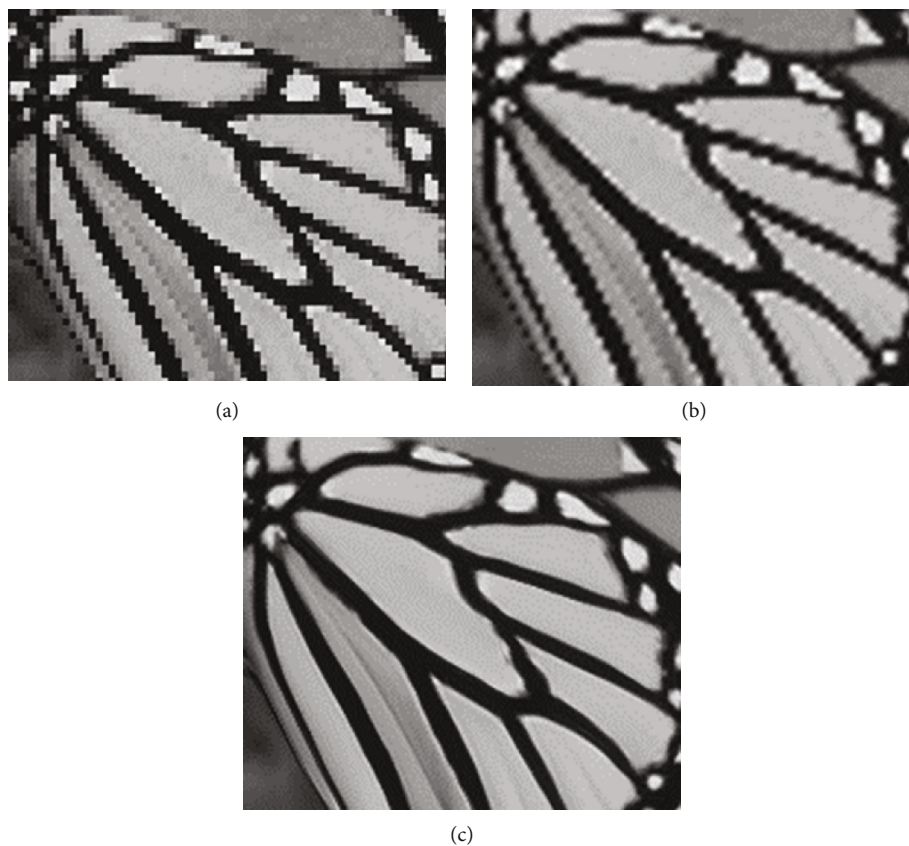


FIGURE 11: LR image *butterfly* and the 3×3 interpolation results: (a) LR image; (b) bicubic (PSNR = 21.77); (c) proposed (PSNR = 23.45).

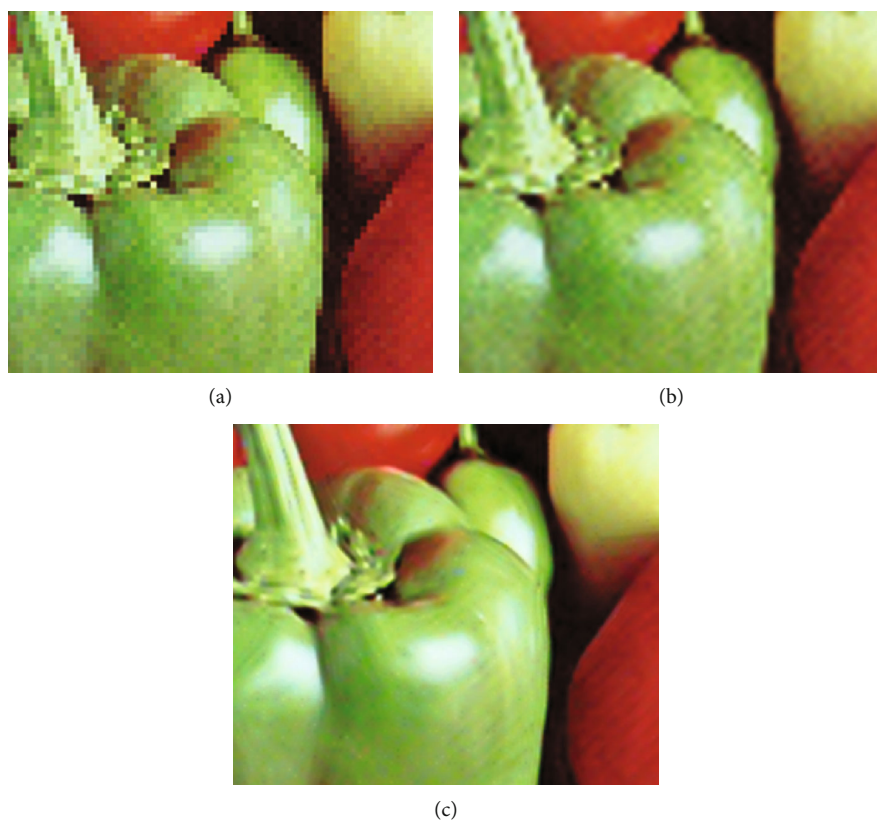


FIGURE 12: LR image *peppers* and the 3×3 interpolation results: (a) LR image; (b) bicubic (PSNR = 30.75); (c) proposed (PSNR = 32.29).

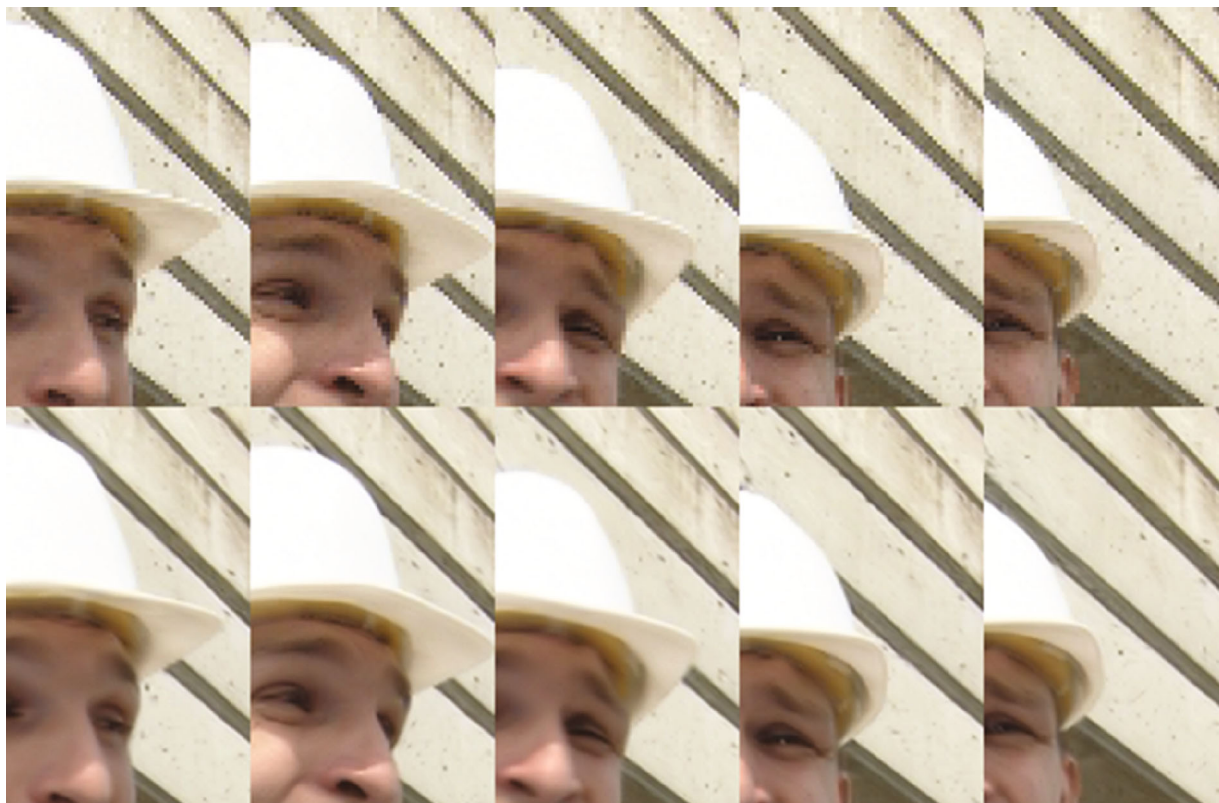


FIGURE 13: LR video sequence *foreman* and the 2×2 interpolation results. Row 1: LR frames (#5, #10, #15, #20, and #25). Row 2: the corresponding HR output frames.



FIGURE 14: LR video sequence *ice* and the 2×2 interpolation results. Row 1: LR frames (#5, #10, #15, #20, and #25). Row 2: the corresponding HR output frames.

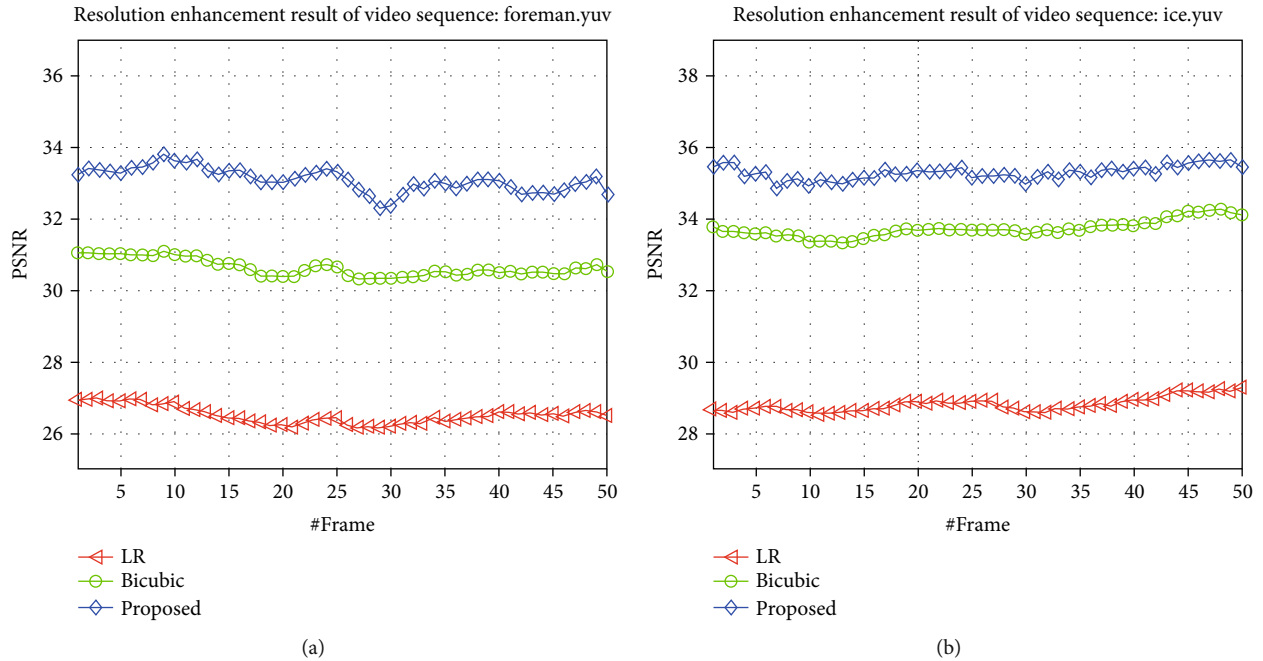


FIGURE 15: The PSNR evaluation of the 2×2 resolution enhancement outputs: (a) *foreman* (average PSNR = 32.88); (b) *ice* (average PSNR = 35.22).

resolution enhanced frame can be obtained by converting these channels back to RGB color space. The diagram is shown in Figure 8.

4. Experimental Results

In this section, several experimental results of the proposed resolution enhancement algorithm are reported to show the performance and compared with the widely used bicubic interpolation method, in terms of subjective image quality and objective PSNR index. The LR input image/video frame is generated by directly decimating the original HR one by a factor of T in each axis and then interpolated back to the original size for performance evaluation. The chosen parameters are as follows: $P = 2$, $n = 8$, $\lambda = 1$, $\gamma = 800$, $\mu = 5$, $M = 12$, $C = 4$, and $W = 20$, the candidate angle set for main direction searching is $\Theta = \{0, 10, 20, \dots, 170\}$, and the width of the directional controllable steerable filter is 5 (with Gaussian kernel standard deviation $\sigma = 0.7$). According to our tests, performing a 2×2 interpolation for a single frame costs about 2.3 seconds on Intel Core i7 8750H with 6 cores at 3.9 GHz, Windows 64 bit, Matlab 2017b, accelerated by C-MEX interface in typical settings of $N = 512 \times 512$ and $D = 2$. Using a GPU-accelerated architecture (CUDA or OpenCL) may be helpful to reduce computation time extremely, we shall study this in future research.

Figures 9–12 present the resolution enhancement results on test still images *leaves*, *airplane*, *butterfly*, and *peppers*, with factor $D = 2$ and 3. Figures 13 and 14 further show the 2×2 interpolation results of test video sequences *foreman* and *ice*, with reference frame number $Q = 2$. From these figures, we see that the proposed algorithm works very well in reconstructing image contours and fine details, with few

noticeable staircase artifacts in tiny structures, when compared to the bicubic interpolation method which produces a large amount of aliasing in edges and textures, and thus, the performance is very poor. Moreover, Figure 15 also gives the objective quality evaluation of *foreman* and *ice* for the first 50 frames. As expected, our method achieves satisfying PSNR values (with about 2 dBs higher than bicubic on average); this is consistent with the subjective visual quality shown above.

5. Conclusion

In this paper, we present an effective algorithm for enhancing digital video/still image resolution based on the directional regularization and nonlocal self-similarity structure, where the missing pixels of an image patch can be estimated from its nonlocal neighbors via an adaptive directional filtering operation. The appeal of this work is its simplicity, with no requirement of solving complex optimization equations, and is easily implemented. Experimental results show that the proposed algorithm can effectively improve the digital video quality in terms of clarity and resolution and thus will be of great value in theory and application.

Data Availability

Please contact the first author (sundong@ahu.edu.cn) to obtain the Matlab demo codes.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62071001), the Anhui Natural Science Foundation of China (Nos. 2008085MF192 and 2008085MF183), the Key Science Project of Anhui Education Department of China (Nos. KJ2018A0012, KJ2019A0023, and KJ2019A0022), and the CERNET Innovation Project of China (Nos. NGII20180612, NGII20180312, and NGII20180624).

References

- [1] T. M. Lehmann, C. Gonner, and K. Spitzer, "Survey: interpolation methods in medical image processing," *IEEE Transactions on Medical Imaging*, vol. 18, no. 11, pp. 1049–1075, 1999.
- [2] Z. F. Lu and B. J. Zhong, "Image interpolation algorithm based on prediction gradient," *Acta Automatica Sinica*, vol. 44, no. 6, pp. 1072–1085, 2018.
- [3] D. Sun, Q. Gao, Y. Lu, L. Zheng, and H. Wang, "A high quality single-image super-resolution algorithm based on linear Bayesian MAP estimation with sparsity prior," *Digital Signal Processing*, vol. 35, no. 12, pp. 45–52, 2014.
- [4] S. Mallat and G. Yu, "Super-resolution with sparse mixing estimators," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2889–2900, 2010.
- [5] B. D. Jiang, Z. Xie, and L. Wu, "Multiresolution visualization of coastline using controllable fractal interpolation," *Journal of Computer-Aided Design & Computer Graphics*, vol. 29, no. 11, pp. 2015–2022, 2017.
- [6] B. Hou, K. Zhou, and L. Jiao, "Adaptive super-resolution for remote sensing images based on sparse representation with global joint dictionary model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 4, pp. 2312–2327, 2018.
- [7] Z. Gao, L. Ding, and C. Xiong, "Single image interpolation using texture-aware low-rank regularization," *Chinese Journal of Electronics*, vol. 27, no. 2, pp. 374–380, 2018.
- [8] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?," *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.
- [9] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.
- [10] S. Izadpanahi and H. Demirel, "Motion based video super resolution using edge directed interpolation and complex wavelet transform," *Signal Processing*, vol. 93, no. 7, pp. 2076–2086, 2013.
- [11] W. Chen, Q. C. Tian, J. Liu, and Q. Wang, "Nonlocal low-rank matrix completion for image interpolation using edge detection and neural network," *Signal, Image and Video Processing*, vol. 8, no. 4, pp. 657–663, 2014.
- [12] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 9, pp. 891–906, 1991.
- [13] D. Sun, Q. Gao, and Y. Lu, "A MAP approach for image deblurring based on sparsity prior and Laplacian mixture modeling," in *2017 32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 901–906, Hefei, China, May 2017.
- [14] W. Dong, L. Zhang, R. Lukac, and G. Shi, "Sparse representation based image interpolation with nonlocal autoregressive modeling," *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1382–1394, 2013.
- [15] D. Sun, Q. W. Gao, and Y. X. Lu, "Image interpolation via collaging its non-local patches," *Digital Signal Processing*, vol. 49, no. 2, pp. 33–43, 2016.
- [16] D. Sun, *Image Interpolation Based on Fractal Self-Similarity and Non-local Patches Collaging*, Anhui University, Hefei, 2016.
- [17] D. Sun, F. Li, and Q. Gao, "A dynamical analysis on non-local autoregressive model and its application on image reconstruction," *Chaos, Solitons & Fractals*, vol. 130, article 109427, 2020.
- [18] M. Elad, *Sparse and Redundant Representation: From Theory to Applications in Signal and Image Processing*, Springer, 2010.