Fast Single Frame Super-Resolution Using Scale-Invariant Self-Similarity

Luhong Liang, King Hung Chiu
Hong Kong Applied Science and Technology Research Institute
Shatin, Hong Kong
{luhongliang, khchiu}@astri.org

Edmund Y. Lam
Department of Electrical and Electronic Engineering
The University of Hong Kong
Pokfulam, Hong Kong
elam@eee.hku.hk

***Abstract*—Example-based super-resolution (SR) attracts great interest due to its wide range of applications. However, these algorithms usually involve patch search in a large database or the input image, which is computationally intensive. In this paper, we propose a scale-invariant self-similarity (SiSS) based super-resolution method. Instead of searching patches, we select the patch according to the SiSS measurement, so that the computational complexity is significantly reduced. Multi-shaped and multi-sized patches are used to collect sufficient patches for high-resolution (HR) image reconstruction and a hybrid weighting method is used to suppress the artifacts. Experimental results show that the proposed algorithm is 20~1,800 times faster than several state-of-the-art approaches and can achieve comparable quality.**

I. INTRODUCTION

Super-resolution (SR) methods aim to recover new high-resolution (HR) information beyond the Nyquist frequency of the low-resolution (LR) image [1]. They attract great practical interest, especially to HDTV, video communication, video surveillance, medical imaging, etc. Recently, much attention has been given to example-based SR using one single LR image, because it can overcome some limitations of the multi-frame SR [2] and can be implemented in lower computation and memory costs.

Example-based SR estimates the missing HR information by capturing the cooccurrence prior between the LR and HR image patches or their transforms. Some standard approaches rely on searching a database built from a representative training image set [3][4][5]. They are capable of producing plausible fine details, but it is hard to efficiently realize such algorithms in hardware. An alternative approach employs the self-similarity characteristics of the natural image. For example, in [6] both the patch recurrence within the same image scale and across different coarser image scales were employed to recover information among subpixel misalignments and implicit LR-HR patch pairs. In [7] a dictionary of LR-HR patch pairs was built online using input LR images of coarser scales and the SR reconstruction relied on Approximate Nearest Neighbor (ANN) searches in the dictionary. In [8] a local self-similarity assumption on natural images were followed, so that patches were extracted from extremely localized regions as small as 10×10 rather than the whole input image. In [9] a local search in a small window was employed to recover non-local redundancy under a back-projection framework. Different from small upscale steps in the aforementioned algorithms, it enlarged the image two times in one step and had lower computation cost.

Example-based SR algorithms are computationally intensive, because for each pixel or patch they need to search the HR counterpart in a database, images or image regions. This has become a major challenge to the commercial application of the SR technology. In this paper, we propose a scale-invariant self-similarity (SiSS) based super-resolution method. Instead of searching the counterpart in the database or image, we use the patch itself for SR reconstruction if it satisfies the characteristics of SiSS, so that the computational complexity is significantly reduced. We use multi-shaped and multi-sized patches to generate sufficient examples for SR reconstruction, and a hybrid weight to suppress the artifacts. The proposed method is 20~1,800 times faster than patch searching methods, while achieving comparable image quality.

The rest of this paper is organized as follows: Section 2 introduces the SiSS based SR algorithm. Section 3 presents the experimental verification. Section 4 concludes the paper.

II. THE PROPOSED APPROACH

A. Scale-Invariant Self-Similarity (SiSS)

Self-similarity is a characteristic of natural image where the local visual content tends to repeat itself within and across the scales of the image. Based on this prior, patch recurrence is recovered in the example-based SR by searching similar patches in the original or downsampled input LR image, and then the HR correspondences are used to reconstruct the HR image [6][7][8]. In this paper we consider a special case of the self-similarity, where the local visual content repeats itself at every scale, namely *Scale-invariant Self-Similarity (SiSS)*. Suppose an image patch P represents a certain local content. We can measure the SiSS of patch P by

$$siss = \min_{s}\left(F\left(\mathrm{C}_{s}(\mathrm{P}),\mathrm{D}_{s}(\mathrm{P})\right)\right), \qquad (1)$$

where $\boldsymbol{D_s(\bullet)}$ is the downsample operation of scaling factor s, $\boldsymbol{C_s(\bullet)}$ is the central region of P with the same size as $\boldsymbol{D_s(P)}$, and $\boldsymbol{F(\bullet)}$ is a similarity measurement. A high *siss* value suggests that the local content appears the same in every scale under the similarity function $\boldsymbol{F(\bullet)}$.

We assume that if a local visual content is SiSS in a certain resolution, it will keep the SiSS characteristic in a higher resolution. Similar assumption on edge sharpness (i.e.

This paper is supported by ITC project #ART/128CP, HKSAR

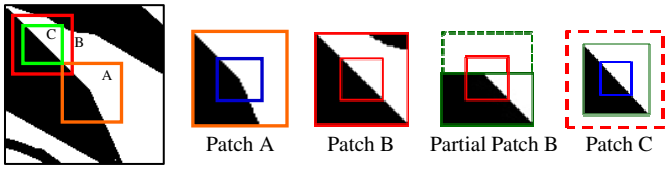


Fig. 1. Example SiSS patch and partial patch.

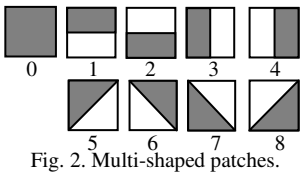


Fig. 2. Multi-shaped patches.

Fig. 3. Patch with "outlier" pixel.

edge prior) has been studied and used in SR [10]. The difference in this paper is that the SiSS is not only a characteristic of the edge but also of more general local structures in the natural image. Based on the assumption, if a patch P has a high *siss*, P and its central region $\boldsymbol{C}_s(P)$ can be considered as a HR-LR pair of the same local visual content. In this case, P can be directly copied to an enlarged image as a HR reconstruction of $\boldsymbol{C}_s(P)$. In other words, instead of searching patches like self-similarity based method, we can measure the SiSS of local visual contents and select those eligible ones to reconstruct the HR contents. Since there is no patch search involved in the reconstruction, the computational complexity could be extremely low.

In practice, we relax the SiSS criterion to a specific scale same as the image enlargement factor. Without loss of generality, suppose $P(x,y)$ is a patch of size $W \times W$, the enlargement factor is 2 and a negative mean-square error (MSE) is used as the similarity function. The SiSS measurement can be simplified to

$$siss = -\underset{-\frac{W}{4}\le x,y\le\frac{W}{4}}{\mathrm{MSE}}\left(\mathrm{P}(x,y),\mathrm{P}_{\downarrow}(x,y)\right), \qquad (2)$$

where $\boldsymbol{P}_{\downarrow}$ denotes the 1/2-downsampled patch P. If the *siss* is larger than a given threshold, patch P is determined to be SiSS.

B. Multi-Shaped and Multi-Sized Patch Reconstruction

In natural images only a small portion of local visual contents, like edge and corner structures, are SiSS, even if the SiSS criterion is relaxed according to (2). Fig. 1 shows an example, where patch A is SiSS, since its central region marked by the blue bounding box appears the same as the patch A itself. In this example, only some patches along the edges can satisfy SiSS, so that only part of the HR image can be reconstructed. However, if we check patch B, we can see that it is not SiSS because of a darker region at the top right hand corner. If we only take the lower part of B into account, the lower partial patch can satisfy SiSS. In this case, the lower part can be directly copied to reconstruct the HR image. Inspired by the above observations, we use multi-shaped partial patches in the SR reconstruction. In practice, 9 different modes are considered as shown in Fig. 2, where mode 0 is a complete square patch and modes 1~8 are partial patches of various shapes. Here, (2) can be re-written as

$$siss(m) = -\underset{(x,y)\in Mode(m)\cap\{(x,y)|-\frac{W}{4}\le x,y\le\frac{W}{4}\}}{\mathrm{MSE}}\left(\mathrm{P}(x,y),\mathrm{P}_{\downarrow}(x,y)\right), \qquad (3)$$

where *Mode(m)* represents the shaded pixels shown in Fig. 2.

Moreover, as for patch B in Fig. 1, it can be seen that if a smaller patch C inside and at the center of B is considered, it becomes SiSS. Apparently, for a natural image of a given resolution, the smaller the patch, the more patches and partial patches that are SiSS. In practice, we use two different sizes of patches: 8×8 and 4×4, where modes 0~8 are used in 8×8 patches and modes 0~4 are used in 4×4 patches. Section 3 will show that sufficient patches can be collected if the multi-shaped and multi-sized patches are used.

C. Data Fusion

After collecting sufficient patches with SiSS, the high-resolution image is reconstructed by blending those patches. A straightforward method is to calculate the weighted average of the overlapped pixels. In our case, it is reasonable to make the weighting a function of the *siss* calculated by (3). Similar to [7] and [9], an exponential function of *siss* is used, such that

$$w = \exp(siss / \sigma_P), \qquad (4)$$

where σ_P is a parameter to control the dependency between the weight and the SiSS characteristics.

The negative value of MSE in (3) describes the overall similarity between the patch and its central region. However, in natural image patches there are often some "outlier" pixels that cannot be described by the overall similarity. Fig. 3 shows a patch with a dark pixel at the top right corner and the zoom-in view of its central region. In this case, it will get a large *siss* value and weight by (3) and (4). Obviously, the patch with high weight will cause artifacts similar to "halo" or ringing in the reconstructed image. Similar examples of "outlier" pixels can be found in partial patches.

To solve this problem, we consider the pixel-to-pixel difference between the patch and its central region by adding a new exponential function into (4), i.e.

$$w(x,y) = \exp(\frac{siss}{\sigma_P})\cdot\exp(-\frac{\left|\left(\mathrm{P}(x,y) - \mathrm{P}_{\downarrow}(u,v)\right)\right|}{\sigma_I}), \qquad (5)$$

where $u = \lfloor x/2 \rfloor$ $v = \lfloor y/2 \rfloor$ and σ_I controls the dependency between the weigtht and the pixel-to-pixel difference. For those "outlier" pixels, the second term in (5) alleviate their impacts to the reconstructed image, so that the artifacts in the reconstructed image can be reduced.

D. Implementation Details

Without loss of generality, we consider the image enlargement with a scale factor of 2. In implementation the LR image is firstly divided into a plurality of overlapping patches. Then, for each patch the *siss* values of multiple modes are calculated. If the *siss* exceeds the threshold, the patch is copied to the enlarged image and blended with overlapped patches. Finally, a back-projection [7][11] is used to keep consistency between the reconstructed HR image and the LR image.

Fig. 4 gives the algorithm details. In Step 2, a fallback image is built by resizing the LR image. Although the multi-shaped and multi-sized patches are used, in natural images there are always a small portion of complex contents that are not SiSS. In this case the fallback image is used to fill the

Input: LR image L, Threshold t_{4x4} and t_{8x8}, Weighting parameter σ_P and σ_I, Threshold for fallback t_f.
Output: HR image H.
1 Set $H(x,y)$=0 and weight image $W(x,y)$=0;
2 Compute a fallback image $F(x,y)$ by resizing image L;
3 Divide L into 8×8 patches using a sliding window moving in ½-pixel step in X and Y directions;
4 **for** each patch P centered at (x_c, y_c)
5 Calculate $siss_0$ for mode 0 using (2);
6 If $siss_0 > t_{8x8}$, then record patch P_0 and $siss_0$;
7 Calculate $siss_1 \sim_8$ for mode 1~8, select the maximum $siss_m$;
8 If $siss_m > t_{8x8}$, then record patch P_m and $siss_m$;
9 **for** each recorded patch P_i
10 Calculate weight $w_i(x,y)$ using (5) given σ_P and σ_I, and calculate $P'_i(x,y)=w(x,y)P_i(x,y)$;
11 Blending by $H(u,v)=H(u,v)+P'_i(x,y)$and $W(u,v)=W(u,v)+w_i(x,y)$, where $u=2x_c+x$, $v=2y_c+y$;
12 Repeat step 2~11 using 4×4 patches and threshold t_{4x4};
13 **for** each pixel at (x,y) in H
14 If w(x,y) > t_f, then normalize by $H(x,y)=H(x,y)/W(x,y)$, else fallback by $H(x,y)=F(x,y)$;
15 Update H using back-projection

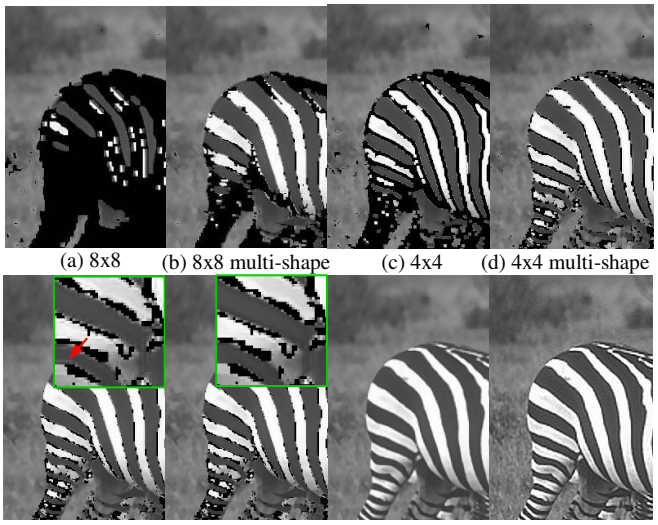
Fig. 4. SiSS based super-resolution algorithm.

"holes" in the reconstructed image in Step 14. In Step 3, the 8×8 patches are collected using a sliding window moving in a ½-pixel step in X and Y directions, so that the patches have been aligned to the grid of the HR image. In practice, the sliding window with ½-pixel displacement can be obtained by edge-guided interpolation [12].

Steps 5~8 describe the multi-shaped patch selection, where we use negative MSE as the similarity function. When the $siss_0$ is calculated for mode 0 in Step 5, all the subtraction and multiplication operations in MSE have been done. Therefore, when the $siss_1 \sim siss_8$ for modes 1~8 are calculated in Step 7, only addition among pixels is required. Since the active pixels in modes 1~8 are rather regular, those additions can be efficiently calculated by an addition tree. The weight calculation in Step 10 is another critical point besides the *siss* calculation. Compared with (4), the proposed weighting method described in (5) has an additional exponential function. For an image of 8-bit pixel depth, there are only 256 possible values of the second term, so that the exponential function can be efficiently realized by a lookup table (LUT).

III. Experimental Results

Firstly, we show the effectiveness of the proposed individual modules. Fig. 5(a) shows the reconstruction result using 8×8 square patches. It can be seen that most of the stripe region cannot be reconstructed due to a lack of SiSS patches. If the multi-shaped patches are used, many more regions can be reconstructed as shown in Fig. 5(b). Fig. 5(c) and (d) show that more regions can be reconstructed using smaller (4×4) patches. Fig. 5(e) shows the result using multi-shaped and multi-sized patches, where most of the regions can be reconstructed. However, there are "halo"-like artifacts around the strong edges, since only the SiSS measurement was used in weighting. If we apply the proposed pixel-to-pixel weighting, the "halo"-like artifacts can be reduced as shown in Fig. 5(f). Fig. 5(g) and (h) show the results after applying fallback image and back-projection (BP).

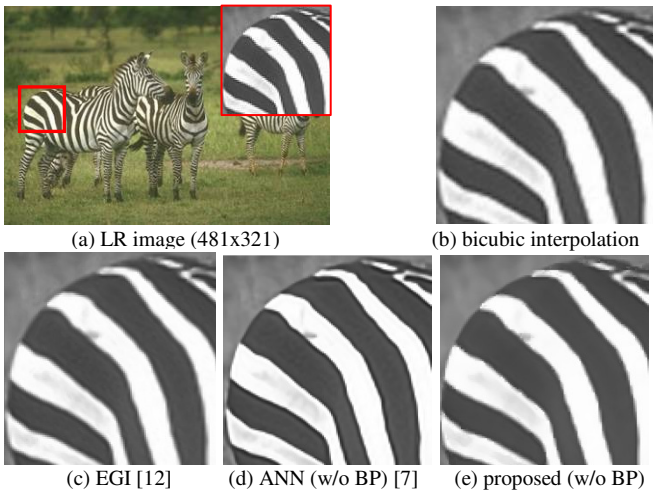


(a) 8x8 (b) 8x8 multi-shape (c) 4x4 (d) 4x4 multi-shape
(e) (a)~(d) all (f) hybrid weight (g) with fallback (h) after BP

Fig. 5. Effectiveness of individual components.

Next, we check the reconstruction quality using a 481×321 image as shown in Fig. 6(a). Fig. 6(b) is the result of the bicubic interpolation, where the edges become blurry and has "jaggy" artifacts. Fig. 6(c) is the result of edge-guided interpolation (EGI) [12], where there is no "jaggy" artifact, but still appears blurry in edges. Fig. 6(d) is the result of ANN based SR [7] without BP, where sharp edges were reconstructed by searching in a large dictionary. Fig. 6(e) shows the result of the proposed algorithm (without BP). It can be seen that although the edge reconstructed is not as sharp as the ANN based SR [7], it is much better than the interpolation methods.

Finally, the overall performance was tested. The LIVE database [13] with 29 high quality images of various scenes was used in the experiments. The original images acted as the ground truth, and the ½-downsampled images using bicubic interpolation acted as the LR images. In patch selection, two different thresholds *10* and *5* were applied to 8×8 and 4×4 patches respectively. In patch weighting σ_P=15 and σ_I=10 were set. The proposed algorithm was implemented in C/C++ without any optimization. The bicubic interpolation, EGI[12], kernel ridge regression [4], sparse coding [5], ANN [7] and



(a) LR image (481x321) (b) bicubic interpolation
(c) EGI [12] (d) ANN (w/o BP) [7] (e) proposed (w/o BP)

Fig. 6. Comparison of reconstruction quality.

NLIBP [9] based methods were tested for comparison. All the programs ran on a desktop with AMD Phenom-II 810 2.60GHz CPU and 4GB memory.

Table 1 lists PSNR and structural similarity (SSIM) [14] scores as well as the average runtimes per image for those algorithms. It can be seen that all the SR algorithms have much better PSNR and SSIM scores, while having higher computational costs than bicubic interpolation. Compared with the listed SR algorithms, the proposed algorithm achieves the best PSNR and SSIM scores, while the runtime is 20~1800 times less.

Table 1. Objective quality comparison[1]

Method	PSNR	SSIM	Time (s)
Bicubic interpolation	26.90	0.952	0.024
Kernel ridge regression [4])	28.64	0.980	101
Sparse Coding [5]	28.45	0.981	1110
ANN (3 BP iterations) [7]	28.46	0.974	1343[2]
NLIBP [9]	28.80	0.969	15.7
Proposed algorithm	28.89	0.991	0.715

The PSNR and SSIM only describe a certain aspect of the image quality, therefore, we also show the subjective quality. Fig. 7 shows results of *Monarch* in the LIVE database. It can be seen that all the SR algorithms outperform the bicubic interpolation and EGI, while the proposed algorithm has comparable quality to other two SR algorithms. Fig. 8 shows an example of one frame in the *Foreman* sequence. It can be seen that the proposed algorithm together with other SR algorithms not just reconstruct sharp edges but recover more detailed information in other regions.

IV. Conclusion

SR is a highly desired technology due to its better quality than most of the interpolation algorithms. However, the high computation cost has become a major obstacle between SR algorithms and real world applications. In this paper, we explored the SR problem in a different point of view. Based on the scale-invariant self-similarity (SiSS) characteristics of the local visual content, we proposed a SR approach based on patch selection rather than patch searching. The proposed multi-shaped and multi-sized patches could create sufficient patches for reconstruction and the improved weighting method suppresses the artifacts. Experimental results showed the effectiveness and efficiency of the proposed algorithm. Future work includes improving the sharpness of the reconstructed edges and suppressing the artifacts more efficiently.

References

[1] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview," IEEE Signal Proces. Mag., vol. 20, no. 3, pp. 21–36, May 2003.

[2] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 9, pp. 1167–1183, Sept. 2002.

[3] W. T. Freeman, T. R. Jones and E. C. Pasztor, "Example-based super-resolution," IEEE Comput. Graph. Appl., vol. 22, pp. 56-65, Mar. 2002.

[4] K. I. Kim and Y. Kwon, "Example-based learning for single-image super-resolution," Pattern Recognition, vol. 5096/2008, pp.456-465, 2008.

[5] J. Yang, J. Wright, T. S. Huang and Y. Ma, "Image super-resolution via sparse representation, " IEEE Trans. Imag. Proc., vol. 19, no. 11, Nov. 2010.

[6] D. Glasner, S. Bagon and M. Irani, "Super-resolution from a single image," Int. Conf. Comput. Vision (ICCV) 2009.

[7] C.-Y. Yang, J.-B. Huang, and M.-H. Yang, "Exploiting self-similarities for single frame super-sesolution," Asian Conf. on Comput. Vision (ACCV), 2010.

[8] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," ACM Trans. Graphics, April 2011.

[9] W. Dong, L. Zhang, G. Shi and X. Wu, "Nonlocal back-projection for adaptive image enlargement," Int. Conf. Imag. Proc. (ICIP), 2009.

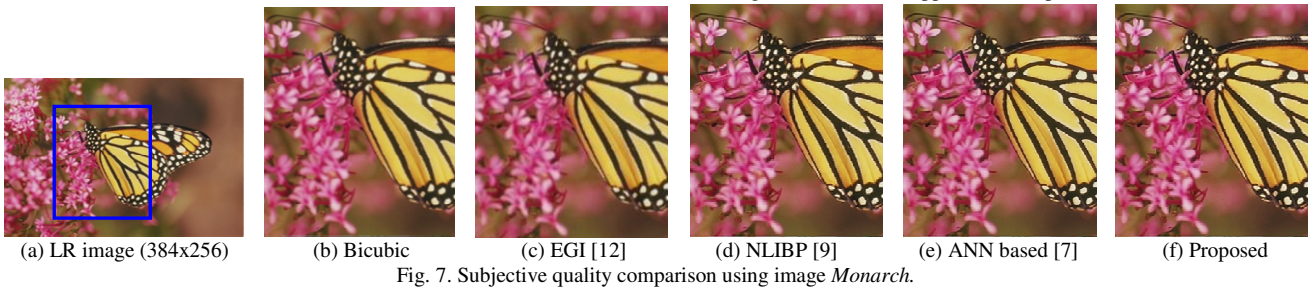
[10] Y.-W. Tai, S. Liu, M. S. Brown and S. Lin, "Super resolution using edge prior and single image detail synthesis," IEEE Conf. Comput. Vision and Pattern Recognition (CVPR) 2010.

[11] M. Irani and S. Peleg, "Improving resolution by image registration," Graphical Models and Imag. Proc. (CVGIP), vol. 53,no.3, pp.231-239, 1991.

[12] L. Zhang and X. Wu, "An edge-guided image interpolation algorithm via directional filtering and data fusion," IEEE Trans. Imag. Proc., vol.15, pp. 2226-2238, Aug. 2006.

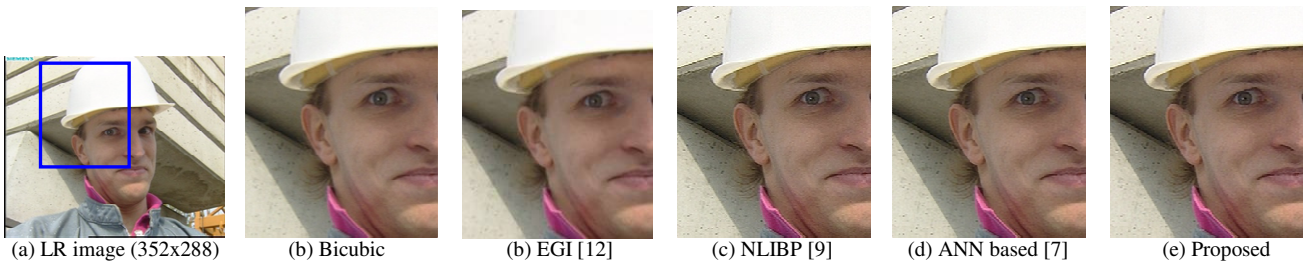
[13] H.R. Sheikh, Z. Wang, L. Cormack, et al., LIVE image quality assessment database 2, (http://live.ece.utexas.edu/research/quality), 2005.

[14] Z. Wang, A.C. Bovik, H.R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Trans. Imag. Proc., vol. 13(4), pp.600-612, April 2004.



(a) LR image (384x256) (b) Bicubic (c) EGI [12] (d) NLIBP [9] (e) ANN based [7] (f) Proposed

Fig. 7. Subjective quality comparison using image *Monarch*.



(a) LR image (352x288) (b) Bicubic (b) EGI [12] (c) NLIBP [9] (d) ANN based [7] (e) Proposed

Fig. 8. Subjective quality comparison using a frame of *Foreman* sequence.

[1] EGI [12] was not tested using PSNR or SSIM due to its different sampling location. [2] Only the runtime of ANN search realized by C/C++ was counted