

RESEARCH

Open Access

Data compression in ViSAR sensor networks using non-linear adaptive weighting



Mohammad R. Khosravi* and Sadegh Samadi

Abstract

Nowadays, industrial video synthetic aperture radars (ViSARs) are widely used for aerial remote sensing and surveillance systems in smart cities. A main challenge of a group of networked ViSAR sensors in an IoT-based environment is low bandwidth of wireless links for communicating big video data. In this research, we propose a non-linear statistical estimator for adaptive reconstruction of compressed ViSAR data. Our proposed reconstruction filter is based on an adaptively generated non-linear weight mask of spatial observations. It can strongly outperform several conventional and well-known reconstruction filters for three different video samples.

Keywords: Video synthetic aperture radar (ViSAR), Non-linear reconstruction filter, Adaptive weighting, Data compression, Interpolation, Internet of Things (IoT)

1 Introduction

The interpolation process is one of the most common processes in remote sensing image and video analysis. Some applications of interpolation in order to estimate unknown pixels are image compression, high-rate video transmission, image and video watermarking, image reconstruction, restoration, and magnification. For instance in [1], a modified scheme was proposed for converting standard-definition television (SDTV) frames to high-definition television (HDTV) standard [2] to be used in video transmission technologies such as DVB-T. Researches on interpolation algorithms include a wide range of research on which some details of them are reviewed as follows. Two most famous interpolators are bi-cubic convolution (mainly abbreviated as BC) and bi-linear (BL) [3]. Today, BC and BL are classified into non-adaptive techniques in terms of local edge computation and indeed provide two linear reconstruction filters [4]. Another main point about them is to use both methods in image processing software tools for remote sensing such as ENVI and ERDAS. However, we wish to focus on newer and efficient types of interpolators entitled edge-guided interpolation algorithms. Edge-guided methods are often applicable in image and video reconstruction problems [5]. In [6], a technique has been

represented which estimates anything based on an assumption that every image can be modeled as a locally stationary Gaussian process. Based on this assumption, the local covariance coefficients in low-resolution (LR) images are estimated, and then, the interpolation process is performed based on geometric duality between the LR and the high-resolution (HR) covariance. A key issue of this method that makes it unsuitable for ViSAR frames is to consider some statistical assumptions which do not exist in practice. In [5], a new scheme was proposed which uses tensor tool for interpolation in order to realize the edge-guided interpolation. This method could outperform some existing methods.

In this research, we want to propose a new edge-guided interpolator based on statistical estimation. Our proposed method uses an adaptive weighting mechanism which makes it edge-guided, non-linear, and fully greedy. Our scheme is an extension for the method discussed in [7–9] for remote sensing applications. In [7], a basic edge-guided interpolation based on linear minimum mean square error estimation (LMMSE) was introduced for benchmark images such that some evaluations about it have been done in [10]. LMMSE includes two phases of directional filtering using a pre-interpolator and data fusion of two orthogonal directions. LMMSE scheme for remote sensing images has been discussed in [9]. This interpolator is a relatively adaptive scheme needing a pre-interpolator based on linear filtering, e.g., linear or cubic interpolation, for directional filtering. In the

* Correspondence: m.khosravi@sutech.ac.ir

Department of Electrical and Electronics Engineering, Shiraz University of Technology, Shiraz, Iran

current work, we are going to propose a full-adaptive version of LMMSE for remote sensing data of ViSAR whereas our proposed technique does not need any pre-interpolator. In fact, if LMMSE can outperform some linear methods like BL or BC, it is completely natural because they have been used as pre-interpolator in LMMSE structure (although as two one-dimensional components), but our proposed method which is named adaptive LMMSE (ALMMSE) can fuse directional observation without need to any linear pre-interpolator and however outperforms the linear interpolators. Our experiments show that it is winner against five conventional techniques among the most popular non-adaptive/linear reconstruction filters.

We can also use the proposed approach for magnifying some multispectral images such as IKONOS and Quick-Bird images or images related to high-resolution optical remote sensing sensors [11–13]. In addition, there are many other applications for interpolation algorithms, e.g., data hiding [14–18], interpolation-based image denoising and demosaicking [19–21], SDTV to HDTV conversion (SD2HD) [2] in video processing, color processing [22], information fusion [8, 9], and shadow detection [23] which can be assisted by ALMMSE algorithm. As we mentioned, the main focus of this research is towards interpolation-based image/video compression [10, 24]. For compression, we firstly down-sample video frames to reduce the information size at the sender side and then reconstruct them using an interpolator at the receiver side. Consequently, ALMMSE can be used in different processes of remote sensing images.

The rest of this paper is organized as follows: in Section 2, we review LMMSE details and some of its applications in digital image processing; then in Section 3, we present our proposed scheme (ALMMSE); and finally, we evaluate it in Section 4. Evaluations show that the use of a locally adaptive estimation in ViSAR frames creates better quality compared to many conventional techniques. The last section is a dedicated conclusion on the work.

2 Related work

LMMSE is a quartered interpolator for creating a four-time larger interpolated image and is widely used in different applications such as enlargement (zooming) [7], noise removal (denoising) [19], color demosaicking [20], and image compression [10]. In this technique, each of non-existing pixels will be computed based on four nearest neighboring pixels which are previously known. Generally, a schematic according to Fig. 1 is used for representing mechanism of LMMSE in two scenarios with orthogonal directions.

The main shortcoming in the design of LMMSE interpolator is to select equal weights for two corresponding pixels which are in the same direction. According to logic of greedy algorithms, LMMSE is not classified into full-adaptive algorithms because it considers a general assumption about generality of images in its computations. However, we can consider it as a partially-adaptive interpolator compared to linear interpolators such as BL and BC. In our study, the aim is to create a new LMMSE-based interpolator for reconstruction of a kind of compressed remote sensing data without need to any pre-interpolation step.

In [21], the authors have proposed an LMMSE-based interpolator for color demosaicking. Demosaicking is a certain type of interpolation which is commonly applicable in some color images, for example Kodak dataset [20]. The demosaicking algorithm in [20] is based on LMMSE and strongly outperforms BL interpolator [3]. Another application of LMMSE is noise reduction. A denoising algorithm is practically a low pass filter which filters high frequency variations of images, or in the other words, it reduces/removes the noises.

Most of the interpolators have mechanisms based on averaging process which is equal to low pass filtering. Therefore, LMMSE-based interpolators can be used in the noise reduction problems. For example in [19], an LMMSE-based denoising algorithm has been proposed for a wide range of digital images. Quality of the scheme is observable.

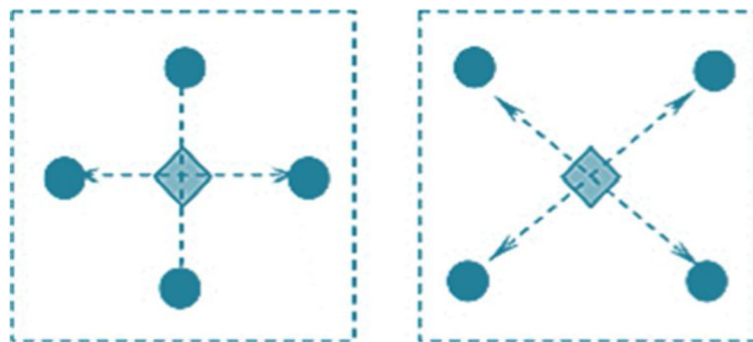


Fig. 1 General positions for underestimate pixels. There are two general positions for underestimate pixels (square symbol) which should be computed by an LMMSE-based interpolator [19]: the left part shows a position with two orthogonal directions of 0 and 90 degrees, and the right shows another position with orthogonal directions of 45 and 135 degrees [7]

Interpolation in spatial domain is also a way for image compression, for more details about LMMSE-based image compression refer to [10]. Thus, in addition to direct applications of interpolators such as magnification, interpolation is widely used for image restoration and reconstruction. In other researches, interpolation-based data hiding [8, 14, 15, 18, 25] and multispectral image fusion (pan-sharpening) [9] are carried out using it. For example in [9], LMMSE has been applied as a magnifier for achieving better quality in pan-sharpening process of Landsat-8 images compared to a linear interpolator. Another application for LMMSE is to do denoising for improving classification accuracy in digital images, because noise reduces accuracy of classifiers (a pre-process based on noise reduction algorithms is normally essential before classification).

3 Proposed method

In the proposed scheme, an interpolation without any assumption regarding estimation weights is applied to

reconstruct compressed frames [4]. In fact, there are no default weights, and all of them are computed adaptively. Our proposed scheme adaptively estimates non-existing pixels to keep edge information in the best way. In this section, we discuss our ALMMSE interpolation method which is an edge-guided scheme and uses four nearest neighbors from two orthogonal directions to estimate targeted pixels; thus, it has suitability for Markov random field (MRF)-based neighborhood systems with order of 1 or 2 such as many remote sensing images. An important point about ALMMSE is to be a full-adaptive non-linear approach that does not need any pre-interpolation compared to linear schemes using polynomials (non-adaptive methods) and traditional LMMSE (with a pre-interpolator).

In order to compress ViSAR frames using the proposed method, we should make down-sampled versions from HR frames (to create LR frames) with lower resolution and then reconstruct the LR frames using our interpolator. To do this, for example, we estimate 75% of

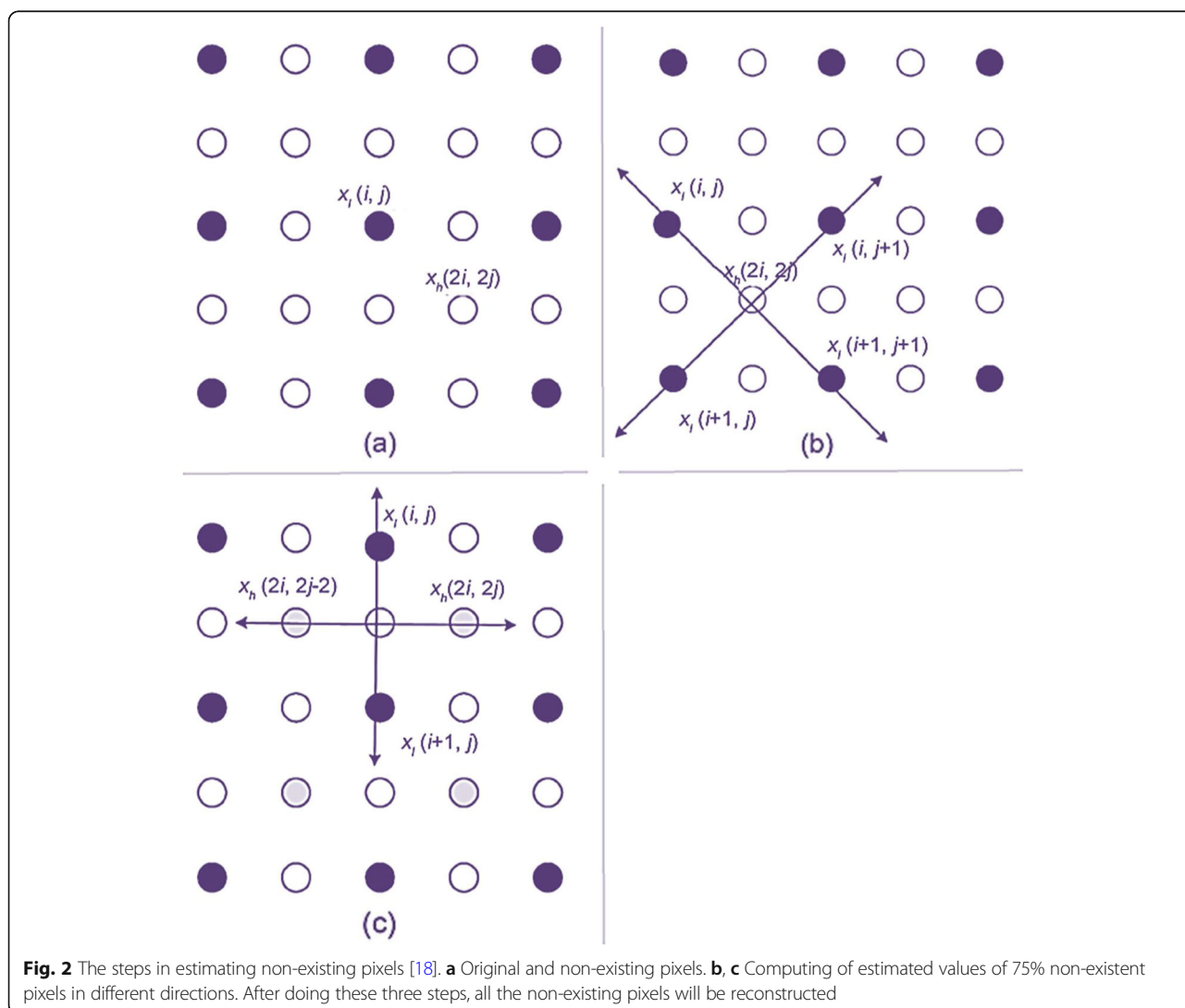


Fig. 2 The steps in estimating non-existing pixels [18]. **a** Original and non-existing pixels. **b, c** Computing of estimated values of 75% non-existent pixels in different directions. After doing these three steps, all the non-existing pixels will be reconstructed

the removed pixels through compression (with a down-sampling algorithm like Algorithm 1) with using only 25% of the remaining cases (sample pixels). In such experiments, we can reduce the video size to be one fourth of the original version. Therefore, we use ALMMSE as a regular interpolation for a quartered template.

Here, we discuss details of ALMMSE interpolation using a template according to Fig. 2a, as seen in continuation of this section. As per Fig. 2 b, five sample pixels are shown, and for simplicity, we utilize simple notations as Eq. (1):

$$\begin{aligned}
 x_l(i, j) &= x_1 \\
 x_l(i, j + 1) &= x_2 \\
 x_l(i + 1, j) &= x_3 \\
 x_l(i + 1, j + 1) &= x_4 \\
 x_h(2i, 2j) &= \hat{x}_h
 \end{aligned} \tag{1}$$

To calculate the non-existing pixels in LMMSE-based interpolated frames, i.e., $\hat{x}_h(x_h)$ is an unknown ideal value and \hat{x}_h is an estimate for x_h and all the same positions, we use a linear combination of original pixels of LR frame. These original pixels are nearest neighbors of the targeted pixel according to Eq. (2) to generate an estimated value (in some scenarios, two of four nearest neighbors are also estimated pixels of a prior step). Although we are using a linear combination, but since all the interpolation weights are specified adaptively and are not fixed, thus, the final reconstruction filter based on ALMMSE will be non-linear [22]. In [22], the traditional LMMSE has been discussed extensively in order to keep the adaptivity for gray levels in every edge area. As can be followed in [22], a general form of LMMSE (for first/second order MRF system and using the simplest pre-interpolator based on two 1D linear estimation (Eq. (3) shows two directional estimates through a kind of bi-linear)) can be written as per Eq. (4). The weights in Eq. (4) are according to Eq. (5) and computable through Eqs. (6–8). Therefore, the ALMMSE closed form is similar as per Eq. (9):

$$\begin{aligned}
 \hat{x}_h &= w_a \hat{x}_a + w_b \hat{x}_b \\
 w_a + w_b &= 1
 \end{aligned} \tag{2}$$

$$\begin{aligned}
 \hat{x}_a &= \frac{x_1 + x_4}{2} \\
 \hat{x}_b &= \frac{x_2 + x_3}{2}
 \end{aligned} \tag{3}$$

$$\begin{aligned}
 \hat{x}_h &= w_a \frac{x_1 + x_4}{2} + w_b \frac{x_2 + x_3}{2} \\
 &= \frac{w_a}{2} x_1 + \frac{w_b}{2} x_2 + \frac{w_b}{2} x_3 + \frac{w_a}{2} x_4
 \end{aligned} \tag{4}$$

$$\{w_a, w_b\} = \underset{w_a + w_b = 1}{\operatorname{argmin}} E \{ (\hat{x}_h - x_h)^2 \} \tag{5}$$

$$w_a = \frac{(x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (\hat{x}_b - \bar{x})^2}{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2 + (\hat{x}_a - \bar{x})^2 + (\hat{x}_b - \bar{x})^2} \tag{6}$$

$$\begin{aligned}
 w_b &= \frac{(x_1 - \bar{x})^2 + (x_4 - \bar{x})^2 + (\hat{x}_a - \bar{x})^2}{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2 + (\hat{x}_a - \bar{x})^2 + (\hat{x}_b - \bar{x})^2} \\
 &= 1 - w_a
 \end{aligned} \tag{7}$$

$$\bar{x} = \frac{1}{4} \sum_{i=1}^4 x_i = \frac{x_a + x_b}{2} \tag{8}$$

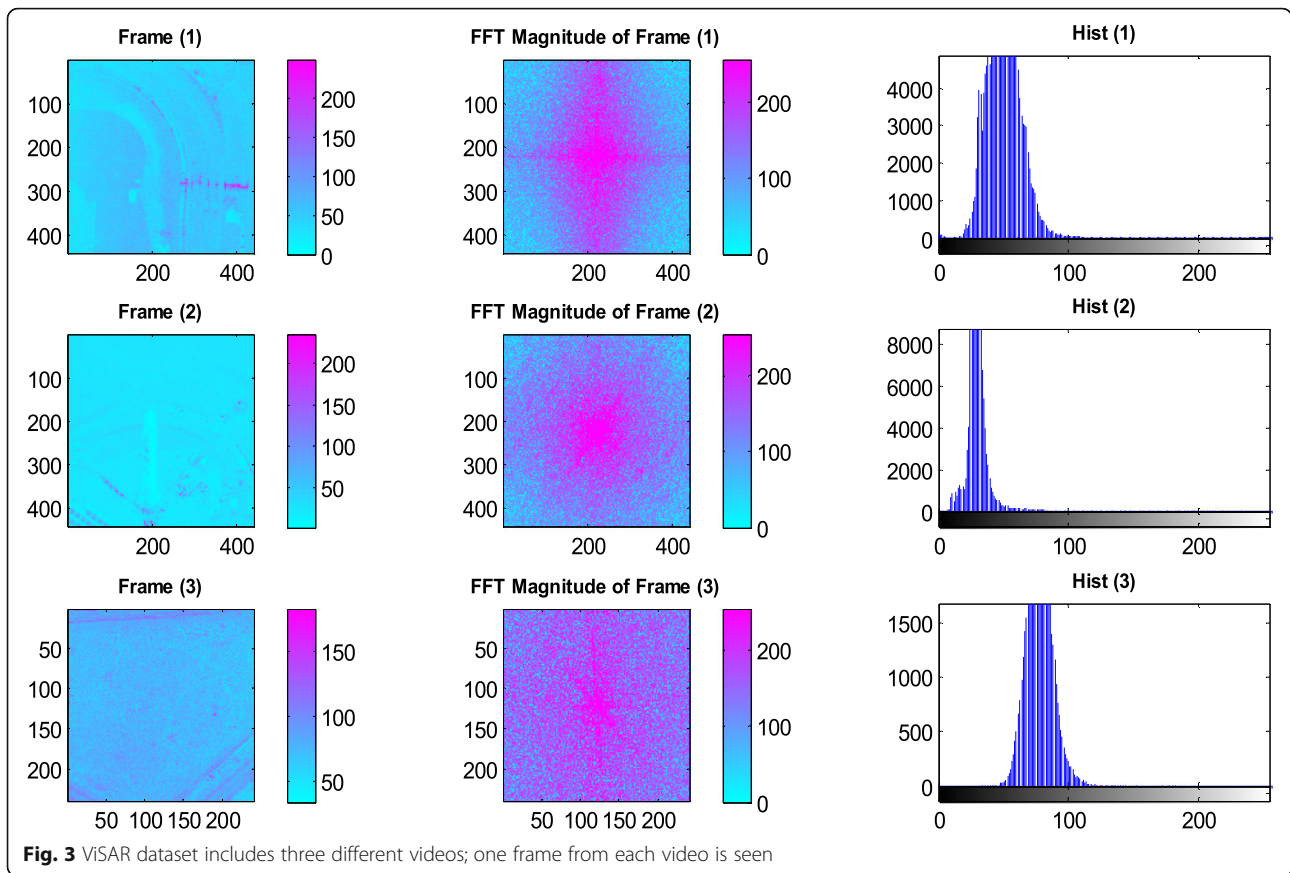
$$\begin{aligned}
 \hat{x}_h &= w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 = \sum_{i=1}^4 w_i x_i \\
 \sum_{i=1}^4 w_i &= 1
 \end{aligned} \tag{9}$$

Now in Eq. (9), we should compute four weights of $w_1, w_2, w_3,$ and w_4 . To do this, there are many ways, but we represent a full-adaptive solution inspired by the traditional LMMSE. Our proposed ALMMSE is indeed a heuristic idea towards extending LMMSE to be full-adaptive (not considering a similar weight for collinear pixels) and with no need to a linear pre-interpolator which makes more computational complexity. For eliminating the pre-interpolation step, we use an approximate as Eq. (10) to make Eqs. (6–7)

Table 1 Qualitative descriptions of the interpolation methods

	Different features in video compression						
	Statistical/ polynomial	Quartered (Q)/non- quartered (NQ)	Quality index	Pre-interpolator needed?	Suitability for SAR videos	Pixel location- based adaptivity	Gray level-based adaptivity (edge-guided)
ALMMSE	Statistical (non-linear)	Q (not flexible)	High	No	Yes	No*	Yes
LMMSE	Statistical (non-linear)	Q (not flexible)	High	Yes	Yes	No*	Yes
Bi-linear	Polynomial (linear)	Q/NQ (flexible)	Adequate	No	Yes	Yes (for NQ)	No
Bi-cubic convolution	Polynomial (linear)	Q/NQ (flexible)	Adequate	No	Yes	Yes (for NQ)	No

*LMMSE/ALMMSE is a quartered interpolator (3:1); thus, it does not need distance-based adaptivity. This adaptation is only applicable for NQ templates



more simple and then generate a heuristic expansion on LMMSE weights to achieve ALMMSE weights (Eq. (11)). As shown in Eqs. (12–16), efficient weights of ALMMSE are very similar to LMMSE weights. In fact, we do not consider any equal weights for these four nearest pixels, and therefore, the approach is fully adaptive whereas LMMSE always selects the same weights for the collinear pixels. In addition, LMMSE has to compute some values as directional estimates of each set of collinear pixels. We could consider an adaptive structure which assumes each pixel as separate sample, regardless of collinearity; therefore, the final approach does not need any pre-estimation for directional estimates that are no longer definable.

$$\begin{aligned} (\hat{x}_a - \bar{x})^2 &\approx 0 \\ (\hat{x}_b - \bar{x})^2 &\approx 0 \end{aligned} \tag{10}$$

$$\{w_1, w_2, w_3, w_4\} = \underset{\sum_{i=1}^4 w_i = 1}{\operatorname{argmin}} E\{(\hat{x}_h - x_h)^2\} \tag{11}$$

The estimation of non-existing pixels will be repeated to achieve all values of 75% of underestimate pixels, as illustrated in Fig. 2c.

$$w_1 = \frac{(x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2}{3((x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2)} \tag{12}$$

$$w_2 = \frac{(x_1 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2}{3((x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2)} \tag{13}$$

$$w_3 = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_4 - \bar{x})^2}{3((x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2)} \tag{14}$$

Table 2 Results for video 1

Schemes	PSNR (dB)	SSIM
Bi-cubic convolution	33.52	0.9377
Bi-linear	33.75	0.9388
Lanczos-2 kernel	33.51	0.9376
Lanczos-3 kernel	33.27	0.9349
Box kernel	32.34	0.9203
ALMMSE	35.04	0.9541

$$w_4 = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2}{3((x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + (x_4 - \bar{x})^2)}$$

$$= 1 - \sum_{i=1}^3 w_i \tag{15}$$

$$\bar{x} = \frac{1}{4} \sum_{i=1}^4 x_i \tag{16}$$

Note that computing of the targeted pixels is firstly based on four nearest neighbors, but in some positions due to the existence of two estimated pixel among four nearest neighbors, practically, the estimation procedure has been performed by six neighbors of which four of these six pixels are not within MRF neighborhood. In the next section, the proposed scheme is evaluated. We will see that the proposed scheme is effective on ViSAR dataset.

Moreover, evaluation is performed based on objective and subjective quality assessment metrics. In addition to the proposed method, a pre-processing step before doing the re-sampling process exists which contains two blocks of down-sampling and up-sampling. Suppose that an input image is a typical $M \times N$ matrix (for simplicity, M and N are even). Algorithm 1 and Algorithm 2 describe these two blocks.

Algorithm 1. Down-sampling Block.

```
% Input: I_{M \times N}
for v = 1:2:M
    for w = 1:2:N
        A((v+1)/2,(w+1)/2) = I(v,w); % Creating the down-sampled matrix A_{\frac{M}{2} \times \frac{N}{2}}
    end
end
% Output: A_{\frac{M}{2} \times \frac{N}{2}}
```

For more details about impacts of different models of down-sampling and up-sampling in quartered interpolators, see detailed discussions in [10]. Table 1 provides more qualitative details of LMMSE, ALMMSE, BL, and BC.

Algorithm 2. Up-sampling Block.

```
% Input: A_{\frac{M}{2} \times \frac{N}{2}}
for v = 1:1:M/2
    for w = 1:1:N/2
        y = (w * 2) - 1;
        x = (v * 2) - 1;
        C(x,y) = A(v,w); % Creating the up-sampled matrix C_{M \times N}
    end
end
% Output: C_{M \times N}
```

4 Results

For evaluation, some ViSAR frames are used which are observable in Fig. 3. PSNR and SSIM as main metrics are used in all evaluations. PSNR and SSIM definitions are seen in Eq. (17) and Eq. (18), respectively, for two

Table 3 Results for video 2

Schemes	PSNR (dB)	SSIM
Bi-cubic convolution	35.33	0.8862
Bi-linear	35.63	0.8869
Lanczos-2 kernel	35.31	0.8859
Lanczos-3 kernel	35.02	0.8810
Box kernel	33.98	0.8521
ALMMSE	36.93	0.9149

entire 8-bit images x and y with the same size $N_1 \times N_2$. In Eq. (18), u_x and u_y denote mean of images, σ_x^2 and σ_y^2 show variance of them, and σ_{xy} describes the covariance between them. Re-sampling is done with the proposed scheme (ALMMSE) and some conventional methods including BL, BC, Lanczos (with parameter of 2 and 3 as an approximation for sinc function), and box kernel [3, 26]. All methods are similar in terms of not having a pre-interpolation step, and this makes the evaluations fair. In addition towards fairness, the down-sampling processes in all methods are the same, the up-sampling in our method is according to Algorithm 2, and the linear methods are according to the MathWork definition.

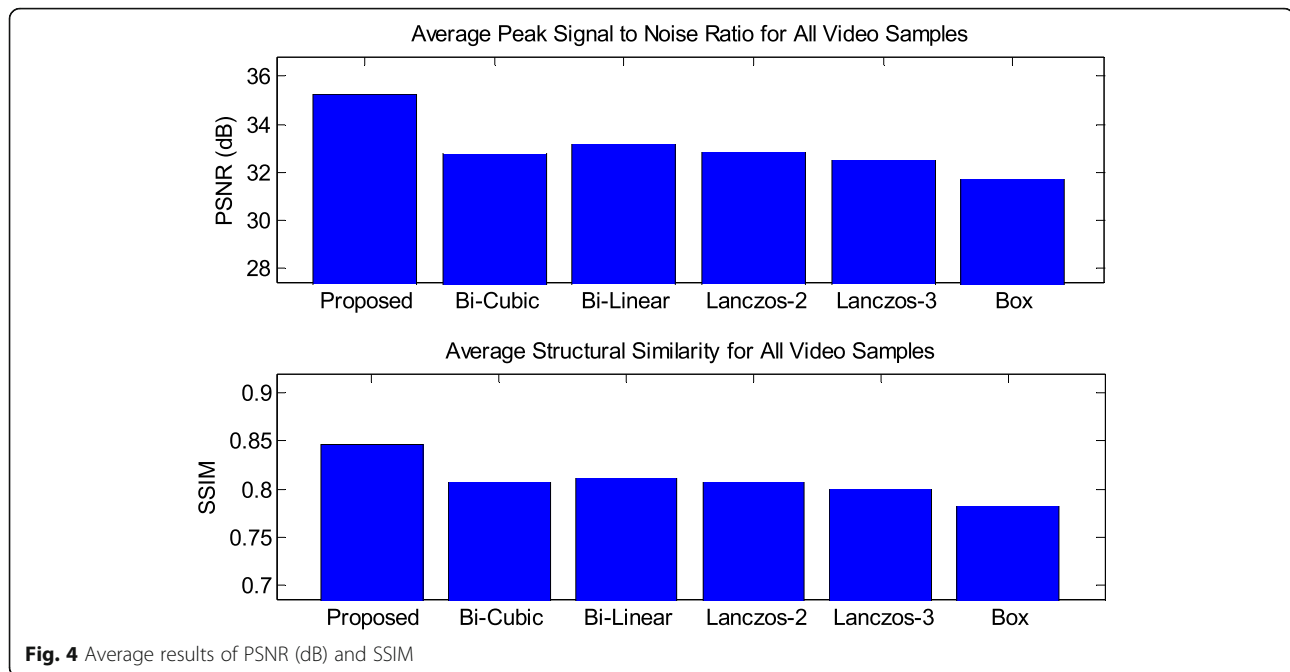
All simulations have been implemented using MATLAB and show that our scheme is strongly winner against non-adaptive/linear methods. Outputs of PSNR and SSIM with complete details are listed in Tables 2, 3, and 4, and Fig. 4 describes an average for all videos. We observe that the proposed scheme based on a full-adaptive approach causes a suitable impact in real ViSAR dataset and outperforms the other methods.

$$PSNR = 20 \log \frac{2^8 - 1}{\sqrt{\frac{1}{N_1 \times N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (x_{ij} - y_{ij})^2}} \tag{17}$$

$$SSIM = \frac{2u_x u_y}{u_x^2 + u_y^2} \times \frac{2\sigma_x \sigma_y}{\sigma_x^2 + \sigma_y^2} \times \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{18}$$

Table 4 Results for video 3

Schemes	PSNR (dB)	SSIM
Bi-cubic convolution	29.49	0.5972
Bi-linear	30.02	0.6045
Lanczos-2 kernel	29.46	0.5958
Lanczos-3 kernel	29.11	0.5824
Box kernel	28.72	0.5705
ALMMSE	30.76	0.6721



5 Conclusions

In the recent years, data processing for IoT became an interesting topic of research [27–30]. In our study, we proposed ALMMSE interpolation algorithm for the remote sensing ViSAR frames captured by imaging radars in an IoT-enabled radar networks of drones and airplanes [31]. This scheme is a new edge-guided interpolator based on non-linear statistical estimation which has no assumption on local weights and also does not need any pre-interpolator. The main feature of the proposed method is to use the most adaptation in comparison to another edge-guided interpolator and conventional interpolation techniques. We compared it with several linear interpolators which do not need any pre-interpolator too. All experiments illustrate a clear consequence about superiority of the proposed method. As a future work, we can go ahead to propose a more accurate version of ALMMSE with lower computational complexity. Evaluation of this proposed method for other remote sensing devices may determine some future directions.

Abbreviations

ALMMSE: Adaptive LMMSE; BC: Bi-cubic convolution; BL: Bi-linear; HDTV: High-definition television; IoT: Internet of Things; LMMSE: Linear minimum mean square error; MRF: Markov random field; SD2HD: SDTV to HDTV data conversion; SDTV: Standard-definition television; ViSAR: Video synthetic aperture radars

Acknowledgements

We would like to thank Sandia National Laboratory for ViSAR data used as dataset in this research.

Authors' contributions

MK participated in the mathematical design of the proposed method and its computer implementation. SS coordinated the industrial application and raw data preparation, and helped out for the study. MK and SS have completed the first draft of this paper. All authors have read and approved the manuscript.

Authors' information

Not applicable.

Funding

Not applicable.

Availability of data and materials

All the data and computer programs are available.

Competing interests

The authors declare that they have no competing interests.

Received: 5 July 2019 Accepted: 10 October 2019

Published online: 02 December 2019

References

1. J. Lua, M. Xiab, W. Li, An interpolation algorithm using center coordinates of pixels for converting SDTV images to HDTV images. *Optik*. 124, 5251–5253 (2013)
2. R. Dianat, M. Ghanbari, Comparison between various standard definition to high definition image conversion methods. *Recent Advances in Communications and Networking Technology*. 4, 6–15 (2015)
3. Gonzalez, R.C.; Woods, R.E. Digital image processing, 3rd edition, 2008, Prentice Hall, New Jersey.
4. Khosravi, M.R. *et al.* A new statistical technique for interpolation of landsat images. *ICAUCAE 2016*, SID Conference Publications, 2016.
5. A. Baghaie, Z. Yu, Structure tensor based image interpolation method. *International Journal of Electronics and Communications*. 69, 515–522 (2015)
6. X. Li, M.T. Orchard, New edge-directed interpolation. *IEEE Transactions on Image Processing*. 10(10), 1521–1527 (2001)
7. L. Zhang, X. Wu, An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE Transactions on Image Processing*. 15(8), 2226–2238 (2006)

8. M.R. Khosravi, M. Yazdi, A lossless data hiding scheme for medical images using a hybrid solution based on IBRW error histogram computation and quartered interpolation with greedy weights. *Neural Computing and Applications* 30, 2017–2028 (2018)
9. Khosravi, M.R. et al. MRF-based multispectral image fusion using an adaptive approach based on edge-guided interpolation. *Journal of Geographic Information System*, 2017.
10. Khosravi, M.R.; Salari, R.; Rostami, H. Analysis of modeling error in digital image spatial estimation. *International Congress of Electrical Engineering, Computer Science and Information Technology (IT2015)*, No. 3, SBU, 2015, pp. 363–371.
11. M. Huang, C. He, Q. Zhu, Qin, J (Maize seed variety classification using the integration of spectral and image features combined with feature transformation based on hyperspectral imaging, *Applied Sciences*, 2016)
12. D. Song, L. Song, Y. Sun, P. Hu, K. Tu, L. Pan, H. Yang, Huang, M (Black heart detection in white radish by hyperspectral transmittance imaging combined with chemometric analysis and a successive projections algorithm, *Applied Sciences*, 2016)
13. S. Lee, M. Kang, K. Uhm, S. Ko, An edge-guided image interpolation method using Taylor series approximation. *IEEE Transactions on Consumer Electronics*. 62(2), 159–165 (2016)
14. M.R. Khosravi, S. Samadi, Modified data aggregation for aerial ViSAR sensor networks in transform domain. The 25th Int'l Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'19), 2019; Las Vegas. USA.
15. M.R. Khosravi, H. Rostami, S. Samadi, Enhancing the binary watermark-based data hiding scheme using an interpolation-based approach for optical remote sensing images. *International Journal of Agricultural and Environmental Information Systems* 9(2), 53–71 (2018)
16. M. Vaidelys, J. Ragulskiene, P. Ziaukas, Ragulskis (M. Image hiding scheme based on the atrial fibrillation model, *Applied Sciences*, 2015)
17. W. Kuo, S. Kuo, Wu, L (Multi-bit data hiding scheme for compressing secret messages, *Applied Sciences*, 2015)
18. L. Luo, Z. Chen, M. Chen, X. Zeng, Z. Xiong, Reversible image watermarking using interpolation technique. *IEEE Transactions on Information Forensics and Security*. 5(1), 187–193 (2010)
19. L. Zhang, X. Li, D. Zhang, Image denoising and zooming under the linear minimum mean square-error estimation framework. *IET Image Processing*. 6(3), 273–283 (2012)
20. Getreuer, P. Zhang-Wu Directional LMMSE image demosaicking. *Image Processing On Line (I POL Journal)*. 2011.
21. L. Zhang, X. Wu, Color demosaicking via directional linear minimum mean square-error estimation. *IEEE Transactions on Image Processing*. 14(12), 2167–2178 (2005)
22. M.R. Khosravi, S. Samadi, R. Mohseni, Spatial interpolators for intra-frame resampling of SAR videos: a comparative study using real-time HD, medical and radar data. *Current Signal Transduction Therapy* (2019)
23. M.K. Moghimi, H. Pourghassem, Shadow detection based on combinations of hessenberg decomposition and principal component analysis in surveillance applications. *IETE Journal of Research*. 61(3), 269–284 (2015)
24. M.Y. Baig, E. Lai, A. Punchihewa, Compressed sensing-based distributed image compression. *Applied Sciences*. 4, 128–147 (2014)
25. Dai, N.; Feng G.; Zeng Q. Reversible watermarking using adaptive edge-guided interpolation, *KSII Transactions on Internet and Information Systems*, 2011.
26. P. Getreuer, Linear methods for image interpolation. *Image Processing On Line*. 1, 238–259 (2011)
27. R. Zhang, P. Xie, C. Wang, G. Liu, S. Wan, *Classifying transportation mode and speed from trajectory data via deep multi-scale learning*, *Computer Networks* 162, 106861 (2019)
28. Z Gao, HZ Xuan, H Zhang, S Wan, KKR Choo, Adaptive fusion and category-level dictionary learning model for multi-view human action recognition, *IEEE Internet of Things Journal*, 2019.
29. S. Wan, Y. Zhao, T. Wang, Z. Gu, Q.H. Abbasi, K.K.R. Choo, *Multi-dimensional data indexing and range query processing via Voronoi diagram for internet of things*, *Future Generation Computer Systems* 91, 382–391 (2019)
30. S. Wan, Y. Zhang, J. Chen, On the construction of data aggregation tree with maximizing lifetime in large-scale wireless sensor networks. *IEEE Sensors Journal* 16(20), 7433–7440 (2016)
31. B. Bahri-Aliabadi, M. Khosravi, S. Samadi, Frame rate computing in video SAR using geometrical analysis. *The 24th Int'l Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'18)*, pp. 165–167, 2018; Las Vegas. USA.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
