

Performance Evaluations for Super-Resolution Mosaicing on UAS Surveillance Videos

Regular Paper

Aldo Camargo¹, Qiang He^{2,*} and Kannappan Palaniappan³¹ Research and Development at I+T, Lima, Peru² Dept. of Mathematics, Computer and Information Sciences, Mississippi Valley State University, Itta Bena, MS, USA³ Dept. of Computer Science, University of Missouri, Columbia, MO, USA

* Corresponding author E-mail: qianghe2005@gmail.com

Received 4 Nov 2012; Accepted 10 Apr 2013

DOI: 10.5772/56534

© 2013 Camargo et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract Unmanned Aircraft Systems (UAS) have been widely applied for reconnaissance and surveillance by exploiting information collected from the digital imaging payload. The super-resolution (SR) mosaicing of low-resolution (LR) UAS surveillance video frames has become a critical requirement for UAS video processing and is important for further effective image understanding. In this paper we develop a novel super-resolution framework, which does not require the construction of sparse matrices. The proposed method implements image operations in the spatial domain and applies an iterated back-projection to construct super-resolution mosaics from the overlapping UAS surveillance video frames. The Steepest Descent method, the Conjugate Gradient method and the Levenberg-Marquardt algorithm are used to numerically solve the nonlinear optimization problem for estimating a super-resolution mosaic. A quantitative performance comparison in terms of computation time and visual quality of the super-resolution mosaics through the three numerical techniques is presented.

Keywords Super-Resolution, Conjugate Gradient Method, Steepest Descent Method, Levenberg-Marquardt Algorithm, Ill-Conditioned Problems, Video Mosaicing

1. Introduction

An Unmanned Aircraft System (UAS) [1] is an aircraft or ground station that can be either remote controlled manually or is capable of flying autonomously under the guidance of pre-programmed GPS waypoint flight plans or more complex on-board intelligent systems. UAS aircrafts have recently found extensive applications in military reconnaissance and surveillance, homeland security, precision agriculture, wildlife conservation, fire monitoring and analysis and other different kinds of aid during disasters. Through surveillance videos captured by a UAS digital imaging payload over the areas of interest, different UAS missions can be conducted. However, the data analysis of UAS videos is frequently limited by motion blurring, resulting from frame-to-frame movement induced by aircraft rolling, wind

gusting, less than ideal atmospheric conditions, the noise inherent within the image sensors, etc. Therefore, the super-resolution mosaicing of low-resolution UAS surveillance video frames has become a critical requirement for UAS video processing and a pre-step for further effective image understanding.

Given multiple images of a particular scene, multi-frame super-resolution reconstructs a high-resolution image with a resolution above the limits of the camera [2-4]. The super-resolved image should have more details than any of the low-resolution images. Mosaicing is the alignment or stitching of two or more images into a single more informative composition representing a 3D scene [5-6]. Generally speaking, the mosaicing creates a panorama, which is impossible to visualize with only one video frame.

Super-resolution mosaicing combines both multi-frame super-resolution and mosaicing and has a number of applications when surveillance video from UAS or satellite is applied. One clear application is the surveillance of certain areas, even during night, with the use of an infrared (IR) imaging system. The UAS can fly over areas of interest and generate super-resolved mosaics that can be analysed at the ground control station. Other important applications involve the supervision of high voltage transmission lines, oil pipes, highway systems, etc. NASA also uses super-resolution mosaics to study the surface of Mars, the Moon and other planets.

Super-resolution mosaicing has been studied by several researchers. Zomet and Peleg [7] applied overlapping areas within a sequence of video frames to create a super-resolved mosaic. In their method, the SR reconstruction technique proposed in [8] is applied to a strip rather than a whole image. This means that the resolution of each strip is enhanced by the fusion of all the frames that contain that particular strip. The disadvantage is that this method is computationally expensive. Ready and Taylor [9] introduced a Kalman filter to compute the super-resolved mosaic. They added unobserved data to the mosaic using Dellaert's method. Basically, they constructed a matrix for the observed pixels to estimate pixel values. This matrix is constructed using a homography matrix and the point spread function (PSF). The problem is that this matrix is extremely large, so they used a Kalman filter and diagonalization of the covariance matrix to reduce the amount of storage and computation. The drawback of this algorithm is the use of a large matrix and the best results with synthetic data obtain a PSNR of 31.6dB. Simolic and Wiegand [10] developed a method based on image warping. In this method, each pixel from every frame is mapped into the SR mosaic and its grey level value is assigned to the

corresponding pixel in the SR mosaic within a range of ± 0.2 pixel units. The drawback of this method is that it requires that the motion vectors and homography must be highly accurate, which is very difficult for real surveillance videos from UAS. Wang, Fevig and Schultz [11] used the overlapped area within five consecutive frames from a video sequence. Then sparse matrices were applied to model the relationship between the LR and SR frames, which can be solved using maximum a posteriori estimation. To deal with the ill-posed problem of the super-resolution model, they adopted hybrid regularization. The drawback of this method is that several sparse matrices have to be built for every five frames. Therefore, this method is not appropriate for dealing with a real video sequence, which contains thousands of frames, in real time. Pickering and Ye [12] proposed an interesting model for mosaicing and super-resolution of video sequences, where the regularization factor is based on the Laplacian operator. The problem with the Laplacian factor is that it forces spatial smoothness. Therefore, both noise and edge pixels are removed in the regularization process. Arican and Frossard [13] use the Levenberg-Marquardt (LM) algorithm to compute the SR of omnidirectional images. Chung [14] proposed a nonlinear least square solution based on the Gauss-Newton method. The disadvantage of this is that it only works for small images.

Our method combines the ideas of most of these techniques, but it also deals with super-resolution mosaicing in a different manner, which does not require the construction of sparse matrices. Therefore, it is feasible to apply the algorithm to a relatively long image sequence and obtain a video mosaic. In addition, we adopt Huber regularization, which preserves high frequency pixels and then sharp edges are also preserved. Furthermore, we model the super-resolution mosaicing problem in a convex framework [4], which guarantees the convergence of the proposed algorithm.

2. Mathematical modelling

2.1 Observation Model

Assuming that there are K frames of LR images available, the observation model can be represented as:

$$\mathbf{y}_k = \mathbf{D}\mathbf{B}_k\mathbf{W}_k\mathbf{R}[\mathbf{x}]_k + \boldsymbol{\eta}_k = \mathbf{H}_k\mathbf{x} + \boldsymbol{\eta}_k \quad (1)$$

Here, \mathbf{y}_k ($k=1, 2, \dots, K$), \mathbf{x} and $\boldsymbol{\eta}_k$ represent the k^{th} LR image, the part of the real world depicted by the super-resolution mosaic and the additive noise, respectively. The observation model in (1) introduces $\mathbf{R}[\mathbf{x}]_k$, which represents the reconstruction of the k^{th} warped SR image from the original high-resolution data \mathbf{x} . The geometric warping operator and the blurring matrix between \mathbf{x} of the real world and the k^{th} LR image frame \mathbf{y}_k are represented by

W_k and B_k , respectively. The decimation operator is denoted with D . The motion between frames is modelled with planar homography. We compute the homography based on the correspondences of SIFT (Scale Invariant Feature Transform) features [15-16] and Random Sample Consensus (RANSAC) strategy [17]. The robustness of the SIFT feature has been verified in feature matching and object recognition. The estimation of the unknown SR mosaic image is not only based on the observed LR images, but also on many other assumptions such as the additive noise and the blurring process. The additive noise, η_k , is considered to be independent and identically distributed (iid) white Gaussian noise. The blurring effect is considered only from the optical equipment. Therefore, the problem of finding the maximum likelihood estimate (MLE) of the SR mosaic image $\hat{\mathbf{x}}$ can be formulated as:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left\{ \left\| \sum_{k=1}^K (\mathbf{y}_k - DB_k W_k R[\mathbf{x}]_k) \right\|^2 \right\} \quad (2)$$

where $\| \cdot \|$ denotes the Euclidean norm. As the SR reconstruction is an ill-posed inverse problem, we need to add another term for regularization, which must contain prior information for the SR mosaicing. This regularization term helps to convert the ill-posed problem into a well-posed solvable problem. Here we adopt the Huber regularization:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left\{ \left\| \sum_{k=1}^K (\mathbf{y}_k - DB_k W_k R[\mathbf{x}]_k) \right\|^2 + \lambda \sum_{g \in G_x} \rho(g, \alpha) \right\} \quad (3)$$

The Huber function is defined as:

$$\rho(x, \alpha) = \begin{cases} x^2, & |x| \leq \alpha \\ 2\alpha|x| - \alpha^2, & \text{otherwise} \end{cases} \quad (4)$$

2.2 Super-resolution Mosaicing Using Steepest Descent Method

Based on the gradient descent algorithm for minimizing (3), the robust iterative update for $\hat{\mathbf{x}}$ can be expressed as:

$$\hat{\mathbf{x}}^{(n+1)} = \hat{\mathbf{x}}^{(n)} + \alpha^{(n)} \left\{ R^T \left[W_k^T B_k^T D^T (\mathbf{y}_k - DB_k W_k R[\hat{\mathbf{x}}^{(n)}]_k) \right]_{k=1}^K - \lambda^{(n)} G^T \rho'(G\hat{\mathbf{x}}^{(n)}, \alpha) \right\} \quad (5)$$

where G is the gradient operator over the cliques [8, 18] and $\lambda^{(n)}$, the regularization operator can be computed as:

$$\lambda^{(n)} = \left(\frac{\sum_{k=1}^K \left\| \mathbf{y}_k - DB_k W_k R[\hat{\mathbf{x}}^{(n)}]_k \right\|^2}{K \sum_{g \in G_x} \rho(g, \alpha)} \right)^2 \quad (6)$$

Furthermore, the derivative of the Huber function is given as:

$$\rho'(x, \alpha) = \begin{cases} 2x, & |x| \leq \alpha \\ 2\alpha \text{sign}(x), & \text{otherwise} \end{cases} \quad (7)$$

The gradient operator G has the advantage over the Total Variation (TV). The Huber function and its gradient with respect to $\hat{\mathbf{x}}^{(n)}$ are continuous as well as convex [19]. Therefore, the optimization problem can be solved using the gradient-descent methods such as the steepest descent and the conjugate gradient methods.

The spatial interactions are adopted in our proposed method. The clique structure determines the spatial interactions, where the activity is computed with finite difference approximations to the second-order directional derivatives (vertical, horizontal and two diagonal directions) in each super-resolution mosaic $\hat{\mathbf{x}}^{(n)}$.

2.3 Super-resolution Mosaicing Using Conjugate Gradient Method

The solution of (3) can be estimated using conjugate gradient as:

$$\hat{\mathbf{x}}^{(n+1)} = \hat{\mathbf{x}}^{(n)} + \beta^{(n)} \mathbf{p}^{(n)} \quad (8)$$

where $\mathbf{p}^{(n)}$ is chosen to be conjugate to all previous search directions with respect to Hessian matrix \mathbf{H} :

$$\mathbf{p}^{(n)} = \nabla f(\hat{\mathbf{x}}^{(n)}) + \left(\frac{\nabla f(\hat{\mathbf{x}}^{(n)})^T \nabla f(\hat{\mathbf{x}}^{(n)})}{\nabla f(\hat{\mathbf{x}}^{(n-1)})^T \nabla f(\hat{\mathbf{x}}^{(n-1)})} \right) \mathbf{p}^{(n-1)} \quad (9)$$

The gradient vector, $\nabla f(\hat{\mathbf{x}}^{(n)})$, is described as follows:

$$\nabla f(\hat{\mathbf{x}}^{(n)}) = R^T \left[W_k^T B_k^T D^T (\mathbf{y}_k - DB_k W_k R[\hat{\mathbf{x}}^{(n)}]_k) \right]_{k=1}^K - \lambda^{(n)} G^T \rho'(G\hat{\mathbf{x}}^{(n)}, \alpha) \quad (10)$$

The gradient operator G is the same as that in the steepest descent method.

2.4 Super-resolution Mosaicing Using Levenverg Marquardt Method

Similar to the gradient methods, the Levenberg-Marquardt method [20] can converge from an initial guess, which may be outside of the convergence region of other methods. In order to minimize (3), we define $\mathbf{u} = f(\mathbf{x})$ as:

$$f(\mathbf{x}) = \left\| \sum_{k=1}^K (\mathbf{y}_k - DB_k W_k R[\mathbf{x}]_k) \right\|^2 + \lambda \sum_{g \in G_x} \rho(g, \alpha) \quad (11)$$

$$f(\mathbf{x} + \delta \mathbf{x}) \approx f(\mathbf{x}) + J \delta \mathbf{x} \quad (12)$$

where $J(\mathbf{x})$ is the Jacobian matrix:

$$J(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \frac{\left\| \sum_{k=1}^K (\mathbf{y}_k - \mathbf{D}\mathbf{B}_k \mathbf{W}_k \mathbf{R}[\mathbf{x}]_k) \right\|^2 + \lambda \sum_{g \in G_x} \rho(g, \alpha)}{\partial \mathbf{x}} \quad (13)$$

$$= \mathbf{R}^T \left[\mathbf{W}_k^T \mathbf{B}_k^T \mathbf{D}^T (\mathbf{y}_k - \mathbf{D}\mathbf{B}_k \mathbf{W}_k \mathbf{R}[\hat{\mathbf{x}}^{(n)}]_k) \right]_{k=1}^K - \lambda^{(n)} \mathbf{G}^T \rho'(\mathbf{G}\hat{\mathbf{x}}^{(n)}, \alpha)$$

The Levenberg-Marquardt method is an iterative process. Initiating at the starting point $\hat{\mathbf{x}}^{(0)}$, the method to solve $\delta \mathbf{x}$ minimizes:

$$\|\mathbf{u} - \mathbf{f}(\hat{\mathbf{x}} + \delta \mathbf{x})\| \approx \|\mathbf{u} - \mathbf{f}(\hat{\mathbf{x}}) - \mathbf{J} \delta \mathbf{x}\| = \|\boldsymbol{\varepsilon} - \mathbf{J} \delta \mathbf{x}\| \quad (14)$$

where $\boldsymbol{\varepsilon} = \mathbf{u} - \mathbf{f}(\hat{\mathbf{x}})$ and then $\delta \mathbf{x}$ can be found by solving a linear least squares problem [18, 20]. The minimum is attained when $\mathbf{J} \delta \mathbf{x} - \boldsymbol{\varepsilon}$ is orthogonal to the column space of \mathbf{J} . This leads to:

$$\mathbf{J}^T \mathbf{J} \delta \mathbf{x} = \mathbf{J}^T \boldsymbol{\varepsilon} \quad (15)$$

$$\mathbf{H}^* \delta \mathbf{x} = \mathbf{J}^T \boldsymbol{\varepsilon} \quad (16)$$

where $\mathbf{H}^* = \mathbf{J}^T \mathbf{J}$ is called the pseudo-Hessian matrix. Then the Levenberg-Marquardt method is to solve Equation (16) by adding a damping term to the diagonal elements of \mathbf{H}^* . The corresponding Levenberg-Marquardt equation is:

$$(\mathbf{H}^* + c\mathbf{I}) \delta \mathbf{x} = \mathbf{J}^T \boldsymbol{\varepsilon} \quad (17)$$

where $\delta \mathbf{x}$ is solved as:

$$\delta \mathbf{x} = \underset{\delta \mathbf{x}}{\operatorname{argmin}} \left\| (\mathbf{H}^* + c\mathbf{I}) \delta \mathbf{x} - \mathbf{J}^T \boldsymbol{\varepsilon} \right\| \quad (18)$$

After $\delta \mathbf{x}$ is known, we have:

$$\hat{\mathbf{x}}^{(n+1)} = \hat{\mathbf{x}}^{(n)} + \delta \mathbf{x} \quad (19)$$

Here c is the Levenberg-Marquardt damping term that determines the behaviour of the gradient in each iteration. If c is close to zero, then the algorithm behaves like a Gauss-Newton (GN) method, but if $c \rightarrow \infty$, then the algorithm behaves like the steepest descent (SD) algorithm. The values of c during the iterative process are chosen in the following way. At the beginning of the iterations, c is set to a large value, so that the LM method integrates the robustness of SD and the initial guess of the solution to (3) can be chosen with less caution. It is necessary to save the errors for each iteration and carry out the comparison between two consecutive errors. If $\text{error}_{(k)} < \text{error}_{(k-1)}$, c is decreased by a certain amount so that LM behaves like the Gauss-Newton method and it speeds up convergence. Otherwise, c is increased to a

larger value, the searching area is then extended, which means that LM behaves like SD. The $\text{error}_{(k)}$ is defined as:

$$\text{error}_{(k)} = \frac{\|\hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k\|}{\|\hat{\mathbf{x}}_k\|} \quad (20)$$

3. Experimental results

The experimental tests are based on three sets of data. One is the synthetic data. The other two are the real UAS data, where one is grey-level image data set and the other is the colour image data set. We created synthetic LR frames from a single high-resolution image. These LR frames were first produced using different translations (18 to 95 pixels), rotations (5° to 10°) and scales (1 to 1.5) and then they were blurred with a Gaussian Kernel. The real grey video data were captured by an experimental small UAS operated by Lockheed Martin Corporation flying a custom-built electro-optical (EO) and an uncooled thermal infrared (IR) imager. The time series of images are extracted from the UAS videos with a low-resolution of 60 × 80. The colour image data are collected with a regular camera mounted in a UAS by Cloud Cap Technology company.

We conducted the three proposed algorithms for super-resolution mosaicing on both synthetic data and real data and then compared their performance. The mosaicing results constructed from the low-resolution input images are set as the initializations for the proposed algorithms. The comparisons are based on PSNR (Peak Signal to Noise Ratio), running time and iteration error for the synthetic data sets and running time and iteration error for the real data from UAS videos because there is no ground truth data available to compute the PSNR for real data. Figures 1, 2 and 3 show the super-resolution mosaics produced from the three different algorithms on the synthetic test data and two sets of real video data. Tables 1, 2 and 3 list the corresponding quantitative comparisons for outcomes from the three different algorithms. From Figures 1, 2, and 3 and Tables 1, 2, and 3, it can be seen that all the methods improve the resolution of the LR mosaic and all of them improve the colour, details and sharpness. However, when the image is grey (IR images), the Levenberg-Marquardt method produces some artefacts since it solves a linear square equation that is close to being singular (13). The final error for the steepest descent and conjugate gradient algorithms decreases with every iteration, which means that they converge to the optimal solution in every step. However, this error from the Levenberg-Marquardt algorithm can decrease or increase due to the use of the damping factor, c , which accelerates the search for the optimal solution. The Levenberg-Marquardt method, interpolating between the Gauss-Newton method and the Gradient Descent method, avoids the time-consuming

computation of the inverse of the pseudo-Hessian matrix in regular singular value decomposition (SVD).

Based on test results on synthetic data and real video data captured from UAS, the Conjugate Gradient method produces the best super-resolution mosaicing results in visual performance. There is almost no difference in

visual performance on the super-resolution of the mosaic images between the Levenberg-Marquardt method and the Steepest Descent method. However, the experimental outcomes show that the Steepest Descent method used the least time among the three approaches to reach the convergence and is the most efficient method.

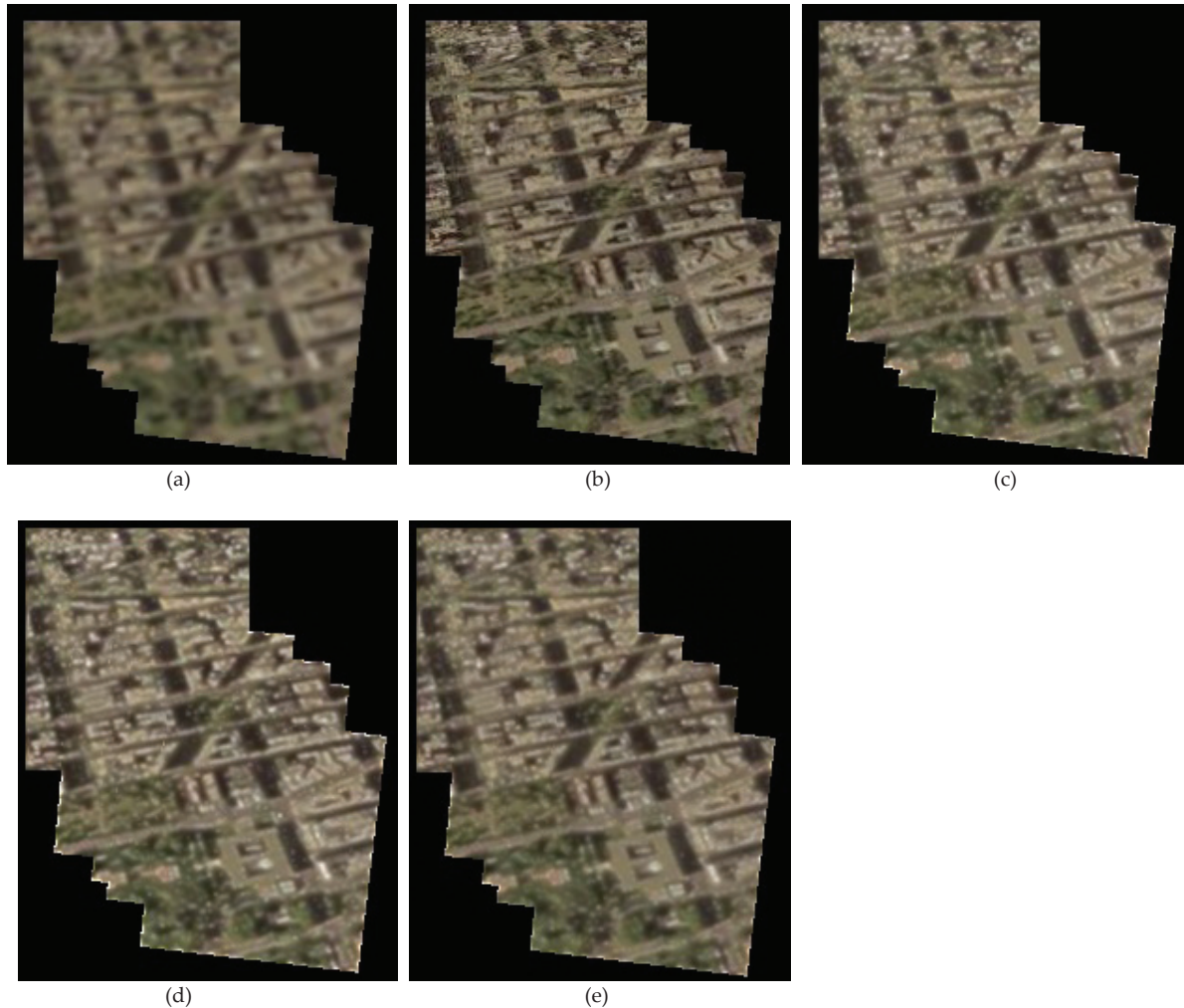


Figure 1. Test on synthetic images. Comparison of the three proposed algorithms: steepest descent, conjugate gradient and Levenberg-Marquardt. (a) LR mosaic. (b) Ground truth HR mosaic. (c) SR mosaic using steepest descent method. (d) SR mosaic using conjugate gradient method. (e) SR mosaic using Levenberg-Marquardt method.

Algorithm	PSNR (dB)	Final error $\text{error}_{(k)} = \frac{\ \hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k\ }{\ \hat{\mathbf{x}}_k\ }$	Total Processing Time on CPU (sec)
Super-resolution using steepest descent algorithm	43.86	0.006391	4.625
Super-resolution using conjugate gradient algorithm	43.98	0.004381	5.047
Super-resolution using Levenberg-Marquardt algorithm	43.77	0.002833	5.422

Table 1. Comparison of the three proposed algorithms to compute super-resolution mosaics for synthetic colour images.

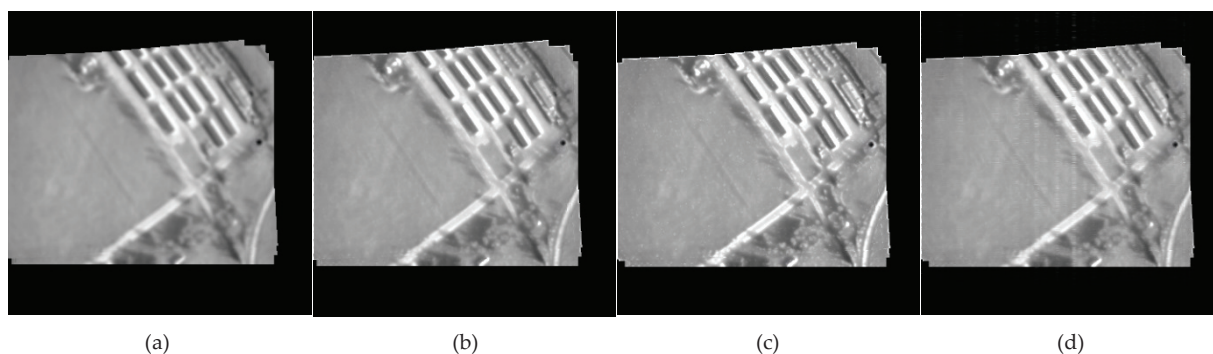


Figure 2. Test on real IR video images captured from UAS. Comparison of the three proposed algorithms: steepest descent, conjugate gradient and Levenberg-Marquardt. (a) LR mosaic. (b) SR mosaic using steepest descent method. (c) SR mosaic using conjugate gradient method. (d) SR mosaic using Levenberg-Marquardt method.

Algorithm	Final error $error_{(k)} = \frac{\ \hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k\ }{\ \hat{\mathbf{x}}_k\ }$	Total Processing Time on CPU (sec)
Super-resolution using steepest descent algorithm	0.065014	10.844
Super-resolution using conjugate gradient algorithm	0.097590	11.907
Super-resolution using Levenberg-Marquardt algorithm	0.068155	11.750

Table 2. Comparison of the three proposed algorithms to compute super-resolution mosaics for real video IR frames captured by UAS.



Figure 3. Test on real colour images captured from UAS. Comparison of the three proposed algorithms: steepest descent, conjugate gradient and Levenberg Marquardt. The images belong to the first set of colour video frames captured from UAS. (a) LR mosaic. (b) SR mosaic using steepest descent. (c) SR mosaic using conjugate gradient. (d) SR mosaic using Levenberg Marquardt.

Algorithm	Final error $error_{(k)} = \frac{\ \hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k\ }{\ \hat{\mathbf{x}}_k\ }$	Total Processing Time on CPU (sec)
Super-resolution using steepest descent method.	0.002469	16.218
Super-resolution using conjugate gradient method.	0.005055	16.891
Super-resolution using Levenberg Marquardt method.	0.005424	17.485

Table 3. Comparison of the three proposed algorithms to compute super-resolution mosaics for real colour video frames captured by UAS.

4. Conclusions

The three optimization methods: the Steepest Descent method, the Conjugate Gradient method and the Levenberg-Marquardt method, are applied to model the super-resolution of the mosaic images. Their running efficiency and visual performance on synthetic test data

and physical test data collected from UAS are compared. Experimentally, the Conjugate Gradient method gives the best super-resolution mosaic results in visual performance while the Steepest Descent method is the most efficient method to converge. There is no large difference in visual performance in the super-resolution

mosaicing from the Levenberg-Marquardt method and the Steepest Descent method.

5. Acknowledgments

This research was supported in part by the Defence Experimental Program to Stimulate Competitive Research (DEPSCoR) and, Army Research Office grant number 50441-CI-DPS, Computing and Information Sciences Division "Real-Time Super-Resolution ATR of UAV-Based Reconnaissance and Surveillance Imagery" (PI, Richard R. Schultz, Principal Investigator,). This research was also supported in part by the Joint Unmanned Aircraft Systems Centre of Excellence, contract number FA4861-06-C-C006, "Unmanned Aerial System Remote Sensing and Avoidance System and Advanced Payload Analysis and Investigation," as well as the North Dakota Department of Commerce grant, "UND Center of Excellence for UAV and Simulation Applications". Additionally, the authors would like to acknowledge the contributions of the Unmanned Aircraft Systems Engineering (UASE) Laboratory team at the University of North Dakota, This research was also supported by Fincyt (Perú) under the SuperRIVAM project.

6. References

- [1] R. K. Barnhart, S. B. Hottman, J. D. D. M. Marshal, E. Shappee, "Introduction to Unmanned Aircraft Systems," 1st ed., CRC Press, October 25, 2011.
- [2] R. R. Shultz, L. Meng and R. L. Stevenson. "Subpixel motion estimation for multiframe resolution enhancement." *Visual Communication and Image Processing* 1997, pp. 1317-1328.
- [3] L. C. Pickup. "Machine Learning in Multi-frame Image Super-resolution," Ph.D. Dissertation, University of Oxford, 2007.
- [4] S. Borman. "Topic in Multiframe Superresolution Restoration," Ph.D. Dissertation, University of Notre Dame, Notre Dame, Indiana, 2004.
- [5] D. P. Capel. "Image Mosaicing and Super-resolution," University of Oxford, Ph.D. Dissertation, 2001.
- [6] S. Farsiu, D. Robinson, M. Elad, P. Milanfar. "Fast and Robust Multi-Frame Super-resolution," *IEEE Transaction on Image Processing*, Vol. 13, No. 10, 2004, pp. 1327-1344.
- [7] A. Zomet and S. Peleg. "Efficient Super-resolution and Applications to Mosaics," *Proc of International Conference of Pattern Recognition*, Sept 2000.
- [8] M. Irani and S. Peleg, "Improving resolution by image registration," *Graph. Models Image Process.*, vol. 53, 1991, pp. 231-239..
- [9] B. B. Ready, C. N. Taylor and R. W. Beard. "A Kalman-filter Based Method for Creation of Super-resolved Mosaics," *Robotics and Automation*, 2006. UCRA 2006.
- [10] A. Smolic and T. Wiegand, "High-Resolution Video Mosaicing," *Proc. ICIP2001, IEEE International Conference on Image Processing*, Thessaloniki, Greece, October 7.-10. 2001.
- [11] Y. Wang, R. Fevig and R. R. Schultz. "Super-resolution Mosaicking of UAV Surveillance Video," *ICIP 2008*, pp. 345-348.
- [12] M. Pickering, G. T. Ye, Michael Frater and Jhon Arnold. "A Transform-Domain Approach to Super-Resolution Mosaicing of Compressed Images," 4th AIP International Conference and the 1st Congress of the IPIA. *Journal of Physics: Conference Series* 124, 2008, 012039.
- [13] Z. Arican and P. Frossard. "Joint Registration and Super-resolution with Omnidirectional Images," *IEEE Transactions on Image Processing*. 2009.
- [14] J. Chung and J. G. Nagy. "Nonlinear Least Squares and Super Resolution," *Journal of Physics: Conference Series* 124, 2008, 012019.
- [15] D. G. Lowe, "Object recognition from local scale-invariant features," in: *Proceedings of the 7th International Conference on Computer Vision*, Kerkyra, Greece, 1999, pp. 1150-1157.
- [16] D. G. Lowe, "Distinctive image features from scale-invariant key points," *International Journal of Computer Vision*, Vol. 60, No. 2, 2004, pp. 91-110.
- [17] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, Vol. 24, No. 6, 1981, pp.381-395.
- [18] W. H. Press, S. A. Teukolsky, A. W. T. Vetterling and B. P. Flannery. "Numerical Recipes in C: The Art of Scientific Computing," Cambridge University Press, New York, 1992.
- [19] R. R. Schultz. "Multichannel Stochastic Image Models: Theory, Applications and Implementations," Ph.D. Dissertation. University of Notre Dame, Indiana.
- [20] D. W. Marquardt. "An Algorithm for the Least-Squares Estimation of Nonlinear Parameters," *SIAM Journal of Applied Mathematics*, Vol. 11, No. 2, 1963, pp. 431-441