2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Fast Local Polynomial Regression Approach for Speckle Noise Removal

Walid K. Sharabati
Department of Statistics
Purdue University
Email: wsharaba@purdue.edu

Bowei Xi
Department of Statistics
Purdue University
Email: xbw@purdue.edu

***Abstract*—In this paper we focus on speckle noise removal. Previously, variational models have been proposed to remove the multiplicative speckle noise. In general, the variational models require a significant amount of run time to converge, and need to set the proper tuning parameter values to achieve optimal noise reduction results. In this paper, we present a local polynomial regression model for speckle noise removal. Our regression model is fast, does not need to be trained on a set of images, does not rely on tuning parameters, and is capable of performing fast speckle noise removal on high resolution images. We have conducted extensive experiments to evaluate our model performance. Our polynomial regression filter outperformed popular noise removal algorithms.**

I. Introduction

Speckle noise is often observed from images produced by coherent imaging systems, such as radar images, ultrasound images, and synthetic aperture radar (SAR) images etc. Speckle noise produces a grainy texture and seriously degrades the image quality. Meanwhile it is a challenging task to remove speckle noise, which is a multiplicative noise. Let $U \in \mathbf{R}_+^n$ denote an n-pixel original image, and $F \in \mathbf{R}_+^n$ be the observed image. Speckle noise is modeled as follows:

$$F = U\eta,$$

where the multiplicative noise η has mean equal to 1 [32], [12]. Speckle noise is often modeled as a Gamma or a Rayleigh distribution [11], [23], [15], [4], [1]. In this article we focus on Gamma distribution. Let $p_n(\eta)$ denote the density of the multiplicative noise.

$$p_n(\eta) = \frac{A^A}{\Gamma(A)}\eta^{A-1}e^{-A\eta},$$

where $\Gamma(\cdot)$ is the Gamma function and $E(\eta) = 1$.

We construct a local polynomial regression filter to remove the speckle noise. For a pixel, the neighboring pixels within radius 5, and the corresponding quadratic and cubic terms are used as predictors in the local polynomial regression filter. Based on Box-Cox transformation, we use the logarithm of the pixel values as the response in the regression filter. Through extensive experiments, we show that our novel local polynomial regression filter achieves outstanding speckle noise removal performance. We summarize the contributions of our approach as follows.

1) Our local polynomial regression filter is a fast approach. The run time is negligible.
2) Our filter does not require any tuning parameter.
3) Our filter does not need a training set of images.
4) Therefore our filter is capable to perform fast speckle noise removal on higher resolution images to achieve the best performance.

This article is organized as follows. In Section I-A we discuss the related work. Sec II presents our fast regression approach for speckle noise removal. In Sec III we conduct extensive experiments to evaluate the performance of our fast local polynomial regression filter and compare with popular denoising algorithms.

A. Related Work

1) Denoising Algorithms for Additive Noise: There is a large literature on removing additive noises, with additive Gaussian noise receiving the most attention. [5] introduced the non-local mean denoising algorithm, where the denoised pixel value is computed by averaging pixels through a Gaussian kernel from neighborhoods with similar structures. Median filter [17] simply uses the median of the neighborhood of a pixel as the output, with different options to define a neighborhood. BM3D filter [9], [8] groups similar neighborhoods, jointly filters the grouped neighborhoods using a Wiener filter, and the final output is a weighted average of the local estimates. [7] applies Wiener filter to patches with similar structure jointly, and the final output is a weighted average. The patch based algorithms [7], [13], [14], [16], through minimizing mean squared error, followed a similar line of reasoning as the least squares type estimators. In [6] a neural network approach has been applied to remove additive noise. [20], [22] focused on additive Poisson noise or additive Gaussian-Poisson noise.

2) Variational Models: A class of variational models, e.g., [29], [2], [32], [12], have been proposed to remove multiplicative noise. Variational models denoise an image $\hat{u}$ by solving a minimization problem, where the target function is the sum of a data fitting term $D(u)$ and a regularization term $R(u)$ multiplied by the regularization parameter $\lambda > 0$. Various variational models used different data fitting terms and different regularization terms, where λ is one of the tuning parameters. Pre-determined tuning parameters significantly affect the variational models performance. Numerically solving such

978-1-5090-4847-2/16/$31.00 ©2016 IEEE

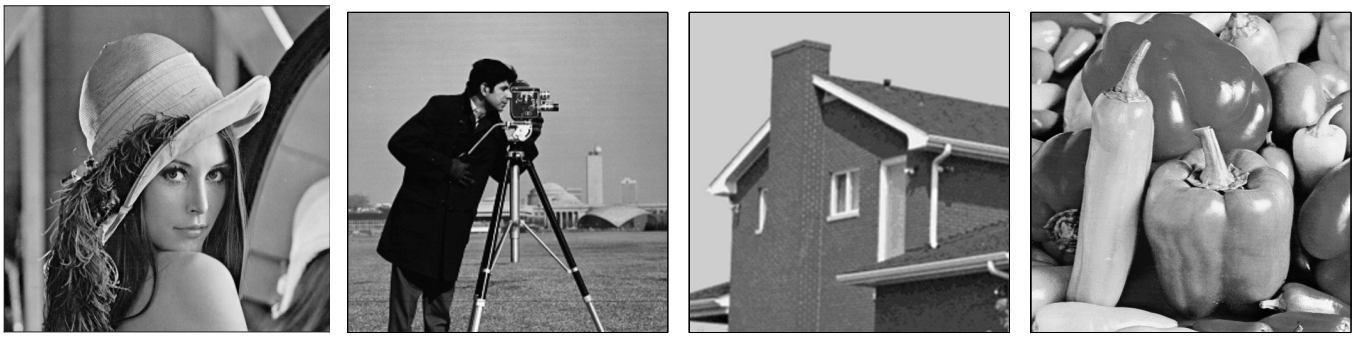

Fig. 1. Clean 512×512 images from left to right: (a) Lenna; (b) Cameraman; (c) House; (d) Peppers

a minimization problem often suffers from slow convergence and subsequently large run time. [32] used Douglas-Rachford splitting technique, which is equivalent in the settings of [32] to the alternating direction methods of multipliers (ADMM), to reduce the computation time needed for numerically solving the problem.

II. Local Polynomial Regression Model

Let $F(i,j)$ be the (i,j)-pixel of the observed image F. We focus on a ball of radius d with $F(i,j)$ at the center. $F(i \pm k, j \pm l)$ with $1 \leq k,l \leq d$ are the neighboring pixels within radius d from $F(i,j)$, $d \geq 1$. Within radius d, there are $4d(d+1)$ neighboring pixels. We construct a local polynomial regression model as follows:

$$\begin{aligned} f(F(i,j)) &= \beta_0 + \sum_{k=1}^{d}\sum_{l=1}^{d} \beta_{1,i-k,j-l} F(i-k,j-l) \\ &+ \sum_{k=1}^{d}\sum_{l=1}^{d} \beta_{1,i+k,j+l} F(i+k,j+l) \\ &+ \sum_{k=1}^{d}\sum_{l=1}^{d} \beta_{2,i-k,j-l} F^2(i-k,j-l) \\ &+ \sum_{k=1}^{d}\sum_{l=1}^{d} \beta_{2,i+k,j+l} F^2(i+k,j+l) \\ &+ \sum_{k=1}^{d}\sum_{l=1}^{d} \beta_{3,i-k,j-l} F^3(i-k,j-l) \\ &+ \sum_{k=1}^{d}\sum_{l=1}^{d} \beta_{3,i+k,j+l} F^3(i+k,j+l) + \epsilon, \end{aligned}$$

where the β's are the coefficients or weights to be estimated by the regression filter.

In our regression model, to obtain a denoised (i,j)-pixel value, the predictors are based on a pixel's neighboring pixels within radius d, a squared neighborhood with the pixel at the center. We have $4d(d+1)$ linear terms, $4d(d+1)$ quadratic terms, and $4d(d+1)$ cubic terms of the neighboring pixels as the predictors. We assume pixels on the edge are reflective and use of the extension theorem to expand on pixels values on and outside the boundary in order to maintain the same number of pixels used in the denoising process.

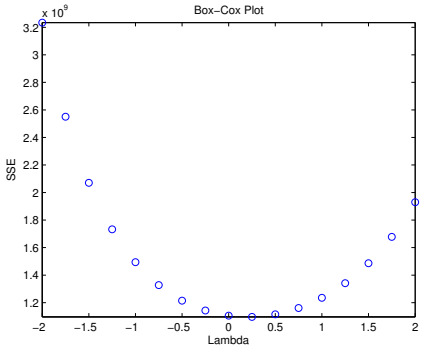

Fig. 6. Box-Cox Transformation for Lenna with $A = 3$

Due to the multiplicative Gamma noise, pixel values $F(i,j)$ are not normally distributed. Therefore we apply Box-Cox transformation to obtain the optimal function of $F(i,j)$ as the response in the regression model. Box-Cox transformation method examines the error sum of squares (SSE) of $F^{\lambda}(i,j)$ for different λ values. For $\lambda = 0$, instead of the power function, Box-Cox method uses $\log F(i,j)$ as the response. Afterwards, the response function with a λ value that returns the smallest SSE is the optimal response. Often the response used in practice has a λ very close to optimal but under a more convenient transformation.

We run Box-Cox transformation for different Gamma A values and different images. The optimal λ values are around 0.25. For example, Figure 6 shows Box-Cox transformation result for Lenna image with $A = 3$. As a standard practice, instead of directly using the optimal λs, we take the value $\lambda = 0$ which is close to optimal, and use the convenient log transformation, $f(F(i,j)) = \log F(i,j)$, as the response in our polynomial regression model. $\log F(i,j)$ is used in Section III.

Through experiments, we discover that a neighborhood of $d = 5$ is sufficient to return an optimal denoised results for various images. Due to space limit, we do not show the experiments for different d values in this paper. Meanwhile, we observe a single pass of our regression filter returns good

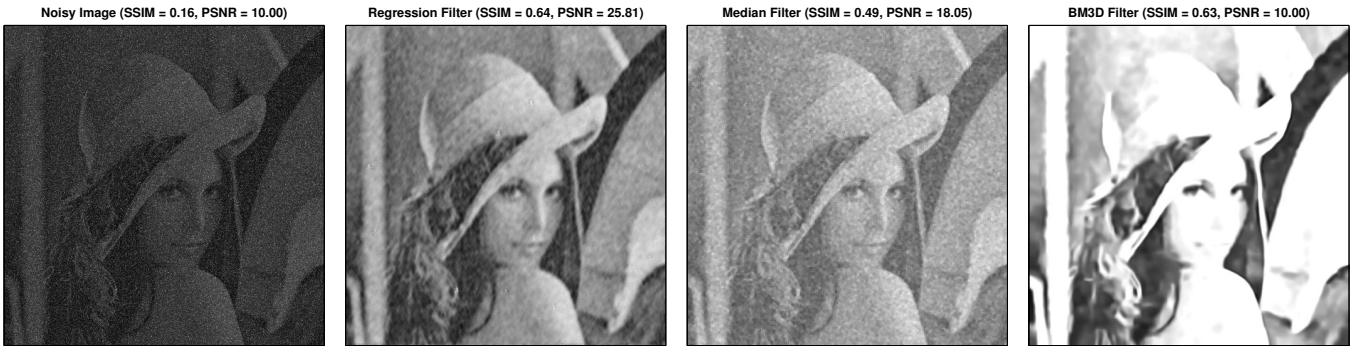

Fig. 2. From left to right: (a) Lenna with speckle noise $A = 3$; (b) Denoised Lenna using our regression approach; (c) Denoised Lenna using median filter; (d) Denoised Lenna using BM3D filter

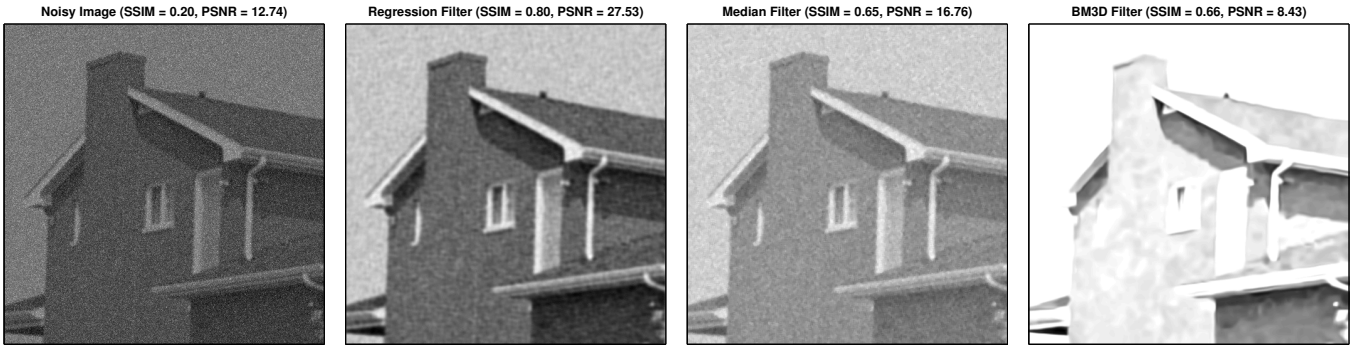

Fig. 3. From left to right: (a) House with speckle noise $A = 9$; (b) Denoised house using our regression approach; (c) Denoised house using median filter; (d) Denoised house using BM3D filter

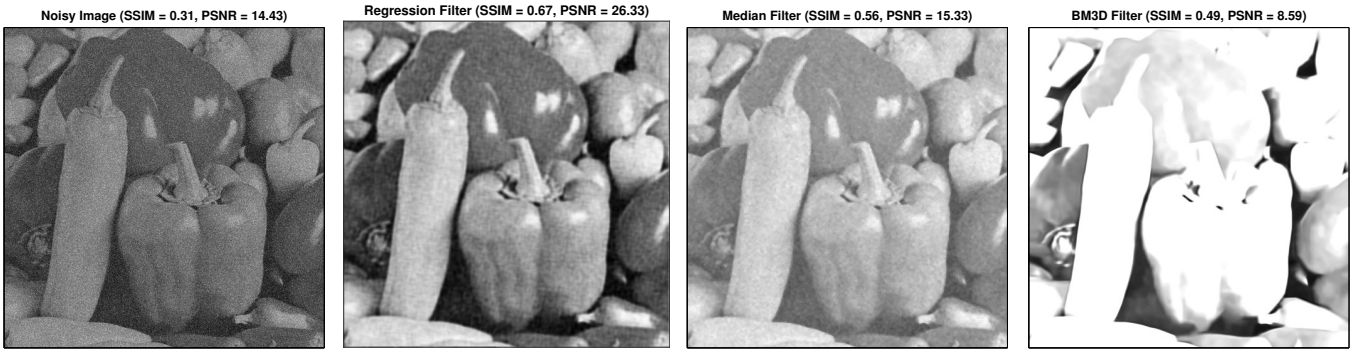

Fig. 4. From left to right: (a) Peppers with speckle noise $A = 15$; (b) Denoised peppers using our regression approach ; (c) Denoised peppers using median filter; (d) Denoised peppers using BM3D filter

results, while 2-3 passes of our regression filter over the noisy images further improve the noise removal results.

We assess the performance of denoising filters with two measures: the peak signal-to-noise ratio (PSNR) [27] and the structural similarity index (SSIM) [31]. In Figures 2, 3, 4, and 5, we show the images with multiplicative speckle noises for four different Gamma A values: 3, 9, 15, and 21. We show the denoised images using our polynomial regression filter, median filter [17] using the Matlab function, and BM3D filter [9]. PSNR and SSIM are displayed atop each image. Figure 1 shows the corresponding clean images.

a) Customized λ: We notice the optimal λ value varies from one image to another due to the inherent structure of an image. We can write the response in our regression filter as $f(F(i,j),\lambda)$, a function of the Box-Cox transformation parameter λ. Based on our collection of general images, we set $\lambda = 0$ and use $\log F(i,j)$ in this article. If a user focuses on a specific application, such as ultrasound images, the user can take a sample of images, run Box-Cox transformation, find the optimal λ, and use another response customized for the application, with the predictors being the linear, the quadratic, and the cubic terms of the neighborhood pixels.

The regression filter is successful in removing speckle

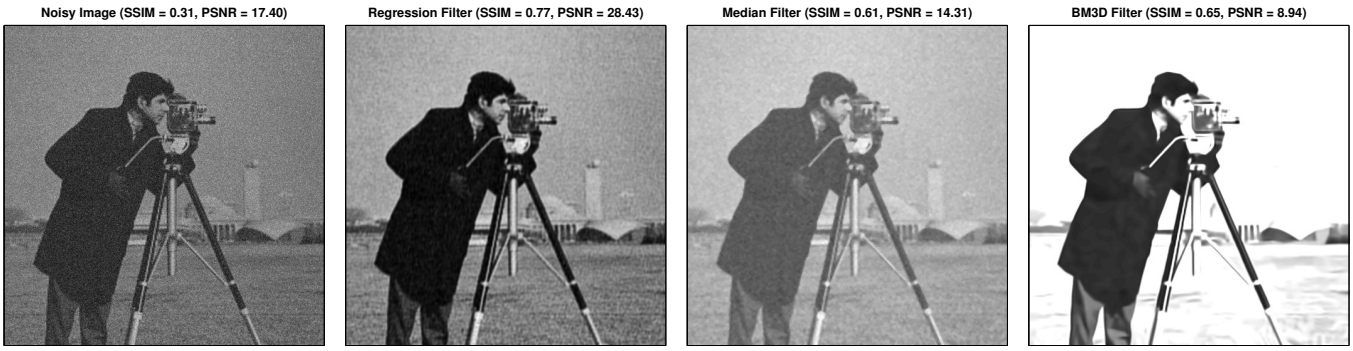

Fig. 5. From left to right: (a) Cameraman with speckle noise $A = 21$; (b) Denoised cameraman using our regression approach; (c) Denoised cameraman using median filter; (d) Denoised cameraman using BM3D filter

noise lies in the facts that we observed a linear association between the noisy pixel and the neighboring pixels and most importantly the fact that regression is robust when it comes to minor violations of the normality assumption.

III. Experiments

A. Experiment One

As shown in Figures 2, 3, 4, and 5, we observe that a speckle noise filter's performance is affected by the inherent structure of the image. Hence we compare our polynomial regression filter with median filter and BM3D filter over 100 images. The images are downloaded from [10]. For every image, we have speckle noise for four different Gamma A values, 3, 9, 15, and 21, run the three filters, and record PSNR and SSIM of the noisy and the denoised images.

Figure 7 shows the difference in PSNR between the noisy images and the denoised images (i.e., PSNR of denoised$-$PSNR of noisy). Figure 8 shows the difference in SSIM between the noisy images and the denoised images (i.e., SSIM of denoised$-$SSIM of noisy). For almost all the images and all the four Gamma A values, our polynomial regression filter significantly outperformed median filter and BM3D filter.

B. Experiment Two

Using 17 images, [12] measured performance of three variational models: the NRSNR model [12], the BF model [3], and the AA model [2]. [12] ran experiments with four Gamma A values: 3, 9, 15, and 21. We downloaded 12 images following the web page link in [12]. The original clean images are 512×512. [12] resized the clean images to 256×256, then multiplied by a Gamma noise, to simplify the parameter tuning process. Our polynomial regression filter is fast, does not have tuning parameters, and is capable of perform speckle noise removal on high resolution images. Hence we conduct two experiments, one with 12 original 512×512 images, and the second one with 12 resized clean 256×256 images. Table I records PSNR of the noisy images and the denoised images using our regression filters for $A = 3$, 9, 15, and 21 based on 512×512 clean images. Table II records PSNR of the noisy images and the denoised images using our regression filters for the same four A values based on 256×256 clean images.

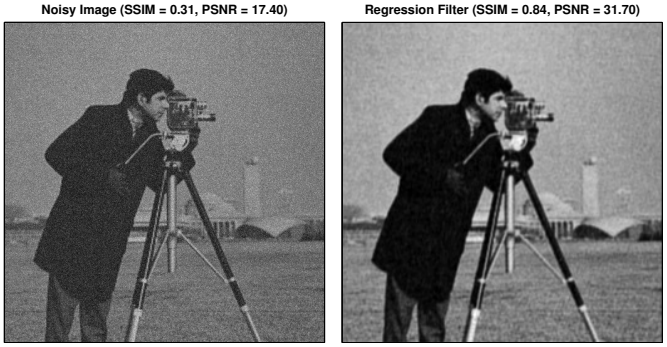

Fig. 9. Left: 512×512 Cameraman with speckle noise $A = 21$; Right: Denoised cameraman using our regression filter and then resized to 256×256

[12] tabled the PSNR results too. For example, house image saw (please refer to [12] for other images): $A = 3$, noisy image PSNR=9.52, NRSNR model PSNR=23.79, (gain of 14.27) with run time 11.66sec; BF model PSNR=23.20, (gain of 13.68) with run time 2.71sec; AA model PSNR=22.46, (gain of 12.94) with run time 14.90sec. $A = 9$, noisy image PSNR=14.24, NRSNR model PSNR=26.73, (gain of 12.49) with run time 17.10sec; BF model PSNR=26.81, (gain of 12.57) with run time 28.78sec; AA model PSNR=25.87, (gain of 11.63) with run time 7.56sec. $A = 15$, noisy image PSNR=16.47, NRSNR model PSNR=28.14, (gain of 11.67) with run time 20.18sec; BF model PSNR=27.81, (gain of 11.34) with run time 29.68sec; AA model PSNR=27.37, (gain of 10.90) with run time 5.58sec. $A = 21$, noisy image PSNR=17.95, NRSNR model PSNR=29.03, (gain of 11.08) with run time 21.39sec; BF model PSNR=28.98, (gain of 11.03) with run time 37.63sec; AA model PSNR=28.26, (gain of 10.31) with run time 4.65sec.

The variational models frequently suffered from large run time in order to converge, and needed to reduce the image resolution to simplify parameter tuning process. On the other hand our polynomial regression filter sees negligible run time and has no tuning parameter. It performs very well and frequently outperforms the variational models as shown in Tables I and II. We also observe that the denoising results

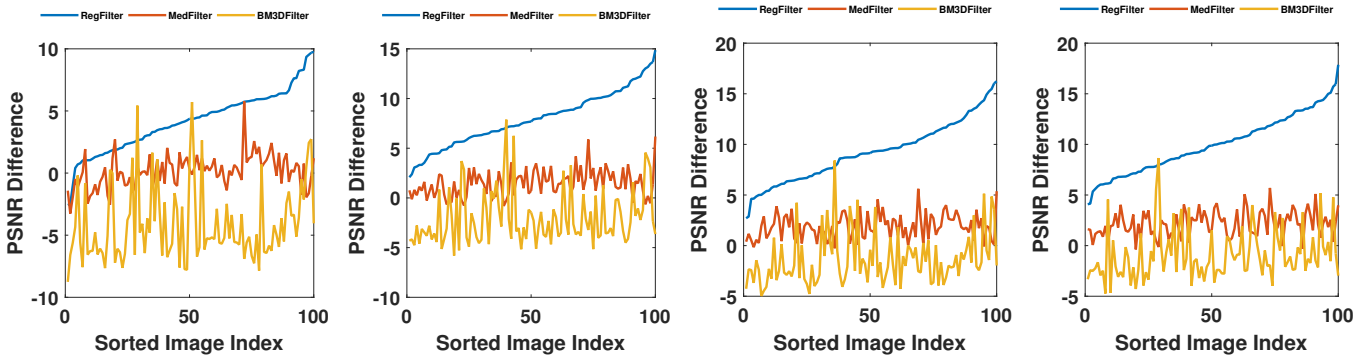

Fig. 7. Difference in PSNR between the noisy image and the denoised image from left to right: (a) $A = 3$; (b) $A = 9$; (c) $A = 15$; (d) $A = 21$

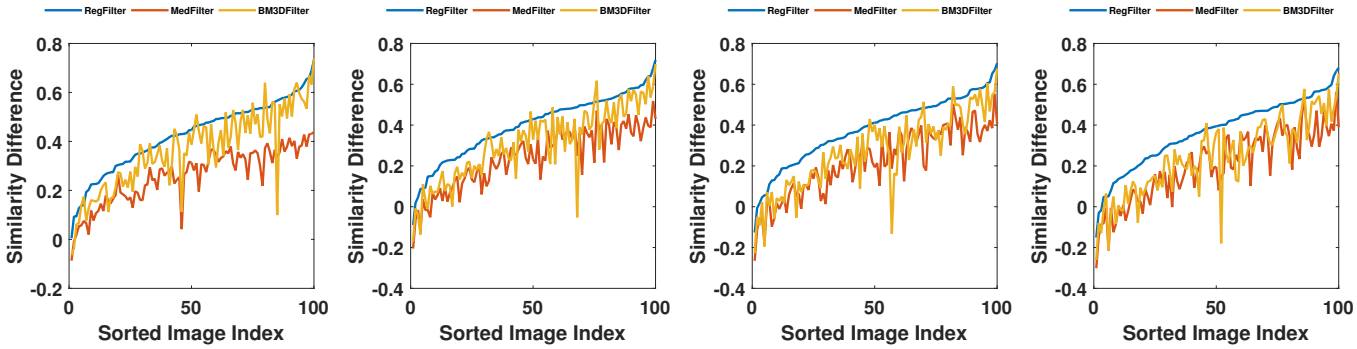

Fig. 8. Difference in SSIM between the noisy image and the denoised image from left to right: (a) $A = 3$; (b) $A = 9$; (c) $A = 15$; (d) $A = 21$

Image	$A = 3$			$A = 9$			$A = 15$			$A = 21$		
	Noisy	Denoised	Gain	Noisy	Denoised	Gain	Noisy	Denoised	Gain	Noisy	Denoised	Gain
Cameraman	10.84	24.99	14.15	15.41	26.77	11.36	17.87	27.79	9.92	17.40	28.44	11.04
House	9.62	27.41	17.79	12.74	27.53	14.79	15.15	27.13	11.98	16.72	29.12	12.4
Jetplane	5.71	18.31	12.6	8.37	24.21	15.84	10.02	25.26	15.24	10.83	25.75	14.92
Lake	8.92	22.39	13.47	11.82	22.10	10.28	13.70	24.18	10.48	14.84	25.11	10.27
Lena	10.00	25.81	15.81	14.61	25.30	10.69	16.59	24.94	8.35	17.94	25.54	7.6
Livingroom	11.82	24.18	12.36	17.16	25.24	8.08	18.73	25.90	7.17	20.48	26.27	5.79
Mandril	9.07	22.04	12.97	12.21	22.85	10.64	13.49	24.14	10.65	17.01	24.72	7.71
Peppers	10.12	24.15	14.03	12.80	25.01	12.21	14.43	26.33	11.90	16.32	25.70	9.38
Pirate	13.30	23.13	9.13	17.50	23.55	6.05	18.72	25.21	6.49	19.15	25.48	6.33
Walkbridge	11.45	21.98	10.53	15.43	22.86	7.43	17.67	22.82	5.15	18.46	23.13	4.67
Blonde	7.05	13.16	6.11	9.85	21.90	12.05	11.96	23.06	11.10	12.81	23.26	10.45
Darkhair	12.74	20.58	7.84	16.03	27.94	11.91	17.00	28.45	11.45	18.03	30.42	12.39

TABLE I
PSNR OF NOISY AND DENOISED IMAGES, AND GAIN IN PSNR, USING OUR REGRESSION FILTER FOR TWELVE 512×512 IMAGES

in general improve significantly on higher resolution images.

b) Resize after denoising a high resolution image: In another experiment, we first perform speckle noise removal on 512×512 images using our polynomial regression filter. Afterwards we resize the denoised images to 256×256, using the Matlab function *imresize()*. We obtain our best denoising results this way. As shown in Figure 9, PSNR increased from 17.40 to 31.70 with a gain of 14.30, and SSIM increased from 0.31 to 0.84 with a gain of 0.53. If only a low resolution image can be displayed, directly performing speckle noise removal on the original high resolution noisy image, then resizing the denoised image to lower resolution is a much better approach. Therefore a fast filter that is capable of handling very high resolution image is critical to achieve good performance.

IV. CONCLUSION

In this article we construct a local polynomial regression filter for speckle noise removal. Through extensive experiments we demonstrate the outstanding performance of our regression filter. Our regression filter is fast and handle higher resolution images with even better performance.

Image	$A=3$			$A=9$			$A=15$			$A=21$		
	Noisy	Denoised	Gain	Noisy	Denoised	Gain	Noisy	Denoised	Gain	Noisy	Denoised	Gain
Cameraman	12.27	23.20	10.93	16.13	24.89	8.76	17.93	25.82	7.89	18.69	26.54	7.85
House	9.97	24.68	14.71	12.99	26.50	13.51	15.28	26.95	11.67	16.03	28.19	12.16
Jetplane	5.63	20.17	14.54	8.29	21.65	13.36	9.96	23.16	13.20	11.02	24.10	13.08
Lake	10.05	21.72	11.70	13.74	22.85	9.11	15.01	23.91	8.90	15.84	25.08	9.24
Lena	10.70	23.44	12.74	14.94	23.54	8.6	17.48	24.65	7.17	18.69	25.16	6.47
Livingroom	11.72	21.98	10.26	16.93	25.11	8.18	18.51	25.78	7.27	20.12	26.31	6.19
Mandril	8.64	20.16	11.52	12.11	22.24	10.13	14.45	22.75	8.30	14.91	22.95	8.04
Peppers	10.75	22.52	11.77	13.24	24.27	11.03	15.45	25.29	9.84	16.50	25.25	8.75
Pirate	13.58	22.09	8.51	16.59	23.64	7.05	17.45	24.40	6.95	18.39	24.52	6.13
Walkbridge	12.45	22.60	10.15	15.74	23.55	7.81	18.22	24.74	6.52	19.12	25.18	6.06
Blonde	7.60	21.65	14.05	11.14	23.72	12.58	12.00	24.47	12.47	12.96	24.44	11.48
Darkhair	13.95	24.44	10.49	16.37	27.24	10.87	17.56	27.27	9.71	18.41	27.55	9.14

TABLE II
PSNR OF NOISY AND DENOISED IMAGES, AND GAIN IN PSNR, USING OUR REGRESSION FILTER FOR TWELVE 256×256 IMAGES

ACKNOWLEDGMENT

This work is partly supported by NSF DMS-1228348.

REFERENCES

[1] M. Afonso, and J. M. Sanches (2015), Image reconstruction under multiplicative speckle noise using total variation. *Neurocomputing*, 150, 200-213.

[2] G. Aubert, and J. F. Aujol (2008). A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4), 925-946.

[3] J. M. Bioucas-Dias, M. A. Figueiredo (2010). Multiplicative noise removal using variable splitting and constrained optimization. *IEEE Transactions on Image Processing*, 19(7), 1720-1730.

[4] E. Bratsolis, M. Sigelle (2003). Fast SAR image restoration, segmentation and detection of high-reflectance regions. *IEEE Trans. Geosci. Remote Sens.* 41(12), 28902899

[5] A. Buades, B. Coll, and J. M. Morel(2005). A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2), 490-530.

[6] H. C. Burger, C. J. Schuler, and S. Harmeling (2012), Image denoising: Can plain neural networks compete with BM3D?. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2392-2399.

[7] P. Chatterjee, and P. Milanfar (2012), Patch-based near-optimal image denoising. *IEEE Transactions on Image Processing*, 21(4), 1635-1649.

[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian (2006), Image denoising with block-matching and 3D filtering. In *Electronic Imaging 2006*, 606414-606414. International Society for Optics and Photonics.

[9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian (2007), Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8), 2080-2095.

[10] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, The PASCAL Visual Object Classes Challenge: A Retrospective, *International Journal of Computer Vision*, 111(1), 98-136, 2015

[11] J. W. Goodman. Statistical Properties of Laser Speckle Patterns, volume 11 of *Topics in Applied Physics*. Springer-Verlag, 2nd ed., 1984.

[12] Y. Han, X. C. Feng, G. Baciu, and W. W. Wang (2013). Nonconvex sparse regularizer based speckle noise removal. *Pattern Recognition*, 46(3), 989-1001.

[13] K. Hirakawa, and T. W. Parks (2006), Image denoising using total least squares. *IEEE Transactions on Image Processing*, 15(9), 2730-2742.

[14] C. Kervrann, and J. Boulanger (2006), Optimal spatial adaptation for patch-based image denoising. *IEEE Transactions on Image Processing*, 15(10), 2866-2878.

[15] H. Lewis. Principle and applications of imaging radar, volume 2 of *Manual of Remote Sensing*. J.Wiley and Sons, 3rd ed., 1998.

[16] A. Levin, and B. Nadler (2011), Natural image denoising: Optimality and inherent bounds. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2833-2840.

[17] J.S. Lim, *Two-Dimensional Signal and Image Processing*, 469-476, Englewood Cliffs, NJ, Prentice Hall, 1990.

[18] M. Liu, Q. Fan (2016). A modified convex variational model for multiplicative noise removal. *Journal of Visual Communication and Image Representation*, 36, 187-198.

[19] J. Liu, T. Z. Huang, I. W. Selesnick, X. G. Lv, and P. Y. Chen (2015). Image restoration using total variation with overlapping group sparsity. *Information Sciences*, 295, 232-246.

[20] F. Luisier, T. Blu, and M. Unser (2011), Image denoising in mixed PoissonGaussian noise. *IEEE Transactions on Image Processing*, 20(3), 696-708.

[21] Z. Qin, D. Goldfarb, and S. Ma (2015). An alternating direction method for total variation denoising. *Optimization Methods and Software*, 30(3), 594-615.

[22] M. Makitalo, and A. Foi (2011), Optimal inversion of the Anscombe transformation in low-count Poisson image denoising. *IEEE Transactions on Image Processing*, 20(1), 99-109.

[23] D. C. Munson and R. L. Visentin (1989). A signal processing view of strip-mapping synthetic aperture radar. *IEEE Transactions on accoustics, speech, and signal processing*, 37(12):21312147.

[24] X. Nie, H. Qiao, and B. Zhang (2015). A Variational Model for PolSAR Data Speckle Reduction Based on the Wishart Distribution. *IEEE Transactions on Image Processing*, 24(4), 1209-1222.

[25] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin (2005). An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2), 460-489.

[26] V. Y. Panin, G. L. Zeng, G. T. Gullberg (1999). Total variation regulated EM algorithm. *IEEE Trans. Nucl. Sci.* 46(6), 22022210

[27] https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio, Peak signal to noise ratio.

[28] M. Rahimi, and M. Yazdi (2015). A new hybrid algorithm for speckle noise reduction of SAR images based on mean-median filter and SRAD method. In *2nd IEEE International Conference on Pattern Recognition and Image Analysis (IPRIA)*, 1-6.

[29] L. Rudin, P. L. Lions, and S. Osher (2003). Multiplicative denoising and deblurring: Theory and algorithms. In *Geometric Level Set Methods in Imaging, Vision, and Graphics*, 103-119, Springer New York.

[30] A. Sawatzky, C. Brune, F. Wbbeling, T. Ksters, K. Schfers, M. Burger (2008). Accurate EM-TV algorithm in PET with low SNR. *IEEE Nuclear Science Symposium Conference Record*, 51335137

[31] https://en.wikipedia.org/wiki/Structural_similarity, Structural similarity index.

[32] G. Steidl, and T. Teuber (2010). Removing multiplicative noise by Douglas-Rachford splitting methods. *Journal of Mathematical Imaging and Vision*, 36(2), 168-184.

[33] C. Wang, L. Xu, D. Clausi, and A. Wong (2015), A Bayesian Joint Decorrelation and Despeckling approach for speckle reduction of SAR Images. *Vision Letters*, 1(1).

[34] J. Yang, J. Fan, D. Ai, X. Wang, Y. Zheng, S. Tang,and Y. Wang (2016), Local statistics and non-local mean filter for speckle noise reduction in medical ultrasound image. *Neurocomputing*, online first version.