Heavy-Tailed Noise Suppression and Derivative Wavelet Scalogram for Detecting DNA Copy Number Aberrations

Nha Nguyen, An Vo, Haibin Sun, and Heng Huang

Abstract—Most existing array comparative genomic hybridization (array CGH) data processing methods and evaluation models assumed that the probability density function (pdf) of noise in array CGH data is a Gaussian distribution. However, in practice such noise distribution is peaky and heavy-tailed. Therefore, a Gaussian pdf is not adequate to approximate the noise in array CGH data and hence introduces wrong detections of chromosomal aberrations and leads misunderstanding on disease pathogenesis. A more accurate and suff cient model of noise in array CGH data is necessary and benef cial to the detection of DNA copy number variations. We analyze the real array CGH data from different platforms and show that the distribution of noise in array CGH data is f tted very well by generalized Gaussian distribution (GGD). Based on our new noise model, we propose a novel array CGH processing method combining the advantages of both smoothing and segmentation approaches. The new method uses generalized Gaussian bivariate shrinkage function and one-directional derivative wavelet scalogram in generalized Gaussian noise. In smoothing step, with the new generalized Gaussian noise model, we derive the heavy-tailed noise suppression algorithm in stationary wavelet domain. In segmentation step, the 1D Gaussian derivative wavelet scalogram is employed to detect break points. Both real and simulated array CGH data with different noises (such as Gaussian noise, GGD noise, and real noise) are used in our experiments. We demonstrate that our new method outperforms other state-of-the-art methods, in terms of both root mean squared errors and receiver operating characteristic curves.

Index Terms—Heavy-tailed noise, wavelet, aCGH, DNA copy number variations.

1 Introduction

Array Comparative Genomic Hybridization (array CGH) is a recently developed technology based on DNA microarrays and dedicated to the investigation and mapping of changes in DNA copy number. Unlike classical CGH with limited resolution (10-20Mb), higher throughput array CGH technology co-hybridizes normal DNA and tumor to a microarray of thousands of bacterial artificial chromosomes, cDNA or oligonucleotide probes, and measures DNA copy number changes in relatively narrow chromosomal regions.

When designing and evaluating chromosomal aberration detection algorithms, most researchers assumed that noise in array CGH follows Gaussian distribution [1], [2], [3], [4], [5]. However, this important assumption has been queried and discussed by [6]. Although they showed that array CGH noise distribution is heavy-tailed, they did not make further conclusion on the kind of distribution. Huang *et. al.* [7] assumed array CGH noise distribution as the Student's t distribution. To address this important problem, in this paper, by considering any deviation from zero values in self-self test samples as noise (the value of true signal is expected as zero over whole sample), we propose a new array CGH noise model, generalized Gaussian distribution (GGD), which covers both Gaussian and heavy-tailed distributions. Five real array CGH data sets with different resolutions and platforms will be studied to support our new noise model assumption. Based on our new noise model, we introduce two new synthetic array CGH data models using either GGD noise or real noise. Hybridization bias problem [8] is also considered in our new synthetic array CGH data models. Compared to traditional models, the new data models generate better simulated array CGH data (closer to real array CGH data) for DNA copy number detection algorithms and evaluations.

To develop effective methods identifying aberration regions from array CGH data, the previous research works mainly utilized one of two key techniques: smoothing and segmentation. In recent work, Lai *et al.* [5] empirically compared 11 different array CGH analysis algorithms and concluded that segmentation-based methods perform consistently well, but when the noise level is high, smoothing-based methods work better.

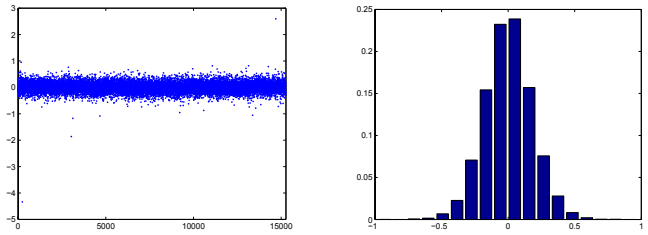
- *N. Nguyen is with the Department of Genetics, Institute for Diabetes, Obesity and Metabolism, School of Medicine, University of Pennsylvania, PA, USA.*
- *A. Vo is with the Feinstein Institute for Medical Research, North Shore LIJ Health System, New York, USA.*
- *H. Sun is with the College of Information Science and Engineering, Shandong University of Science and Technology, China.*
- *H. Huang is with the Department of Computer Science and Engineering, University of Texas at Arlington, TX, USA.*
- *Corresponding Authors: H. Huang, heng@uta.edu; N. Nguyen, nhqnha@gmail.com; H. Sun, sdustsun@163.com*

- *H. Huang is supported by NSF-IIS 1302675, NSF-IIS 1344152, NSF-DBI 1356628, NSF-IIS 1619308, NSF-IIS 1633753, NIH R01 AG049371.*

Smoothing-based methods remove noise in frequency domain to discover low amplitude aberration regions and reduce the number of identified false aberration regions. However, smoothing-based methods cannot detect exactly breakpoints of aberration regions, because changing points in array CGH are corresponding to high frequency component which could be suppressed in denoising process. Segmentation-based methods target to model data as a series of discrete segments under certain optimization criterion and directly give out the final results with visible gain, deletion or normal regions. The segmentation-based methods could more accurately detect the boundary points. However, because the small aberration regions are highly possible to be buried into its neighbors in high noise case, the false positives could be easily introduced. It would be very desirable to develop new methods to process array CGH data with advantages from both smoothing and segmentation approaches [6].

In this paper, we propose a novel derivative wavelet scalogram based segmentation method (DWSS) to identify DNA copy number aberrations by integrating both smoothing and segmentation steps and handling heavy-tailed noise. Our DWSS method includes two main steps: heavy-tailed noise suppression and breakpoint detection. Compared to the state-of-the-art related work, our method has advantages in three folds. 1) Lai *et al.* [5] proposed Wave method using stationary wavelet transform that works well with Gaussian noise. Instead of hard threshold [9], in our work, generalized Gaussian bivariate shrinkage function is designed in stationary wavelet domain to suppress both Gaussian and heavy-tailed noise in array CGH. 2) Ben *et al.* [10] proposed HaarSeg algorithm using simple wavelet based pattern-matching or wavelet footprint to detect breakpoints in array CGH. HaarSeg algorithm run fast and gave promising segmentation results. Inspired by the pattern-matching idea, we propose Gaussian derivative wavelet scalogram to segment pre-processed array CGH. 3) Pique-Regi *et al.* [8], [11] proposed two GADA algorithms in which GADA1 was designed with an unbiased measurement assumption, and GADA2 worked well with probe hybridization bias. It is necessary to have a segmentation method working with both bias and unbias cases. Thus, we also consider the hybridization bias problem and design our method to be robust with bias [8] as well as unbias measurement in array CGH.

We validate our method in both synthetic and real array CGH data sets. Moreover, in synthetic data, we introduce two kinds of data by adding real noise and GGD noise. Both Root Mean Square Errors (RMSE) and Receiver Operating Characteristic (ROC) curves are calculated to evaluate the performance. In all experimental results, our new DWSS method consistently outperforms the state-of-the-art array CGH data processing algorithms, and also shows fast computational time.



(a) Chromosome 15 (b) Empirical histogram

Fig. 1. An example of array CGH GSM232967 and its empirical histogram.

Data set	Number of Arrays	Platform	Number of samples
Lee 2008 array	40	Nimblegen 385K	2
Snijders 2001 array	15	HumArray 1.14	89
Bredel 2005 array	26	Stanford human cDNA microarray	26
Smith 2007 array	69	Agilent 244K	3
Nicolas 2009 array	23	Custom Nimblegen	2
Kidd 2010 array	22	Agilent 244K	1
Perry 2008 array	66	Agilent 244K	3
Bovee 2008 array	66	Agilent 244K	3

TABLE 1
Eight datasets which are used to analyze noise in array CGH with many platforms

2 Array CGH Noise Characteristic

In this section, the array CGH noise distribution will be analyzed using the self-test samples of real array CGH data, in which the deviations from zero values are considered as noise. Several possible candidates of noise models are used to fit the noise and the relative entropy is used to evaluate the fitting performance. Compared with all candidates, the generalized Gaussian distribution (GGD) that covers both Gaussian and heavy-tailed distributions, has the best fitting results on array CGH noise.

2.1 Data Description

We analyze eight real array CGH data such as Lee 2008 array [12], Snijders 2001 array [13], Bredel 2005 array [14], Smith 2007 array [15], Nicolas 2009 array [16], Kidd 2010 array [17], Perry 2008 array [18] and Bovee 2008 array [19] as shown in Table 1. All of them are available to the public. Similar to previous related research [6], we consider the true signal of a normal chromosome should only include copy two, hence deviations values from zero ($\log(2/2) = 0$) in real signal of a normal chromosome are the noise in array CGH. Table 1 describes the details of each data set, including the data platform, the number of self-self test samples, and the number of arrays (chromosomes).

In the Lee 2008 array [12] which used Nimblegen Macaque Whole genome CGH 385K array, there are two

self-self test samples (GSM232967, GSM232968) of log2-transformed ratios (CH1/CH2) with some ten-thousand probes. Totally we have 40 (2 samples $\times$ 20 chromosomes) chromosomes. The Snijders 2001 array [13] is from Stanford University with 15 human cell lines. Each chromosome in this data only contains around one hundred probes. The Bredel 2005 array [14] data is from Harvard Medical School. This data includes 26 samples, and each sample has thousands of probes. With low resolution data, in order to get enough data points for fitting, we will combine many normal chromosomes of the same sample together. The Smith 2007 data [15], using Agilent-015366 Custom Human 244K CGH Microarray, includes three control self-self hybridization samples, and each sample has twenty-three chromosomes. For this data, we have 69 chromosomes with ten-thousand probes each. Kidd 2010 [17], Perry 2008 [18] and Bovee 2008 [19] are three other data sets which also used Agilent-015366 Custom Human 244K CGH Microarray. They have seven self-test samples with 22 chromosomes each. In the Nicolas 2009 [16] which used Custom Nimblegen array CGH chip, we have one self-test sample (GSM334824) of 23 chromosomes in for noise analysis.

2.2 Noise Distribution Candidates for Array CGH

After studying these eight data sets, we found the noise distribution of array CGH is bell-shaped and symmetric. One example is shown in Fig. 1. It is chromosome 15 of GSM232967 in Fig. 1(a) from the Lee 2008 array [12]. Fig. 1(b) is the empirical histogram of the signal in Fig. 1(a). We can see this histogram is bell-shaped and symmetric. There are four probability distribution candidates for such noise, including Gaussian distribution, generalized Gaussian distribution (GGD), Student's t distribution, and Cauchy distribution. There is another bell-shaped distribution, extreme value distribution, but it is not symmetric.

We will focus on four bell-shaped and symmetric distribution candidates as shown in Fig. 2. Most previous works [1], [2], [3], [4], [5], [20] assumed that probability density function (pdf) of noise in array CGH is zero-mean Gaussian shown in Fig. 2(a) as follows:

$$p(x;\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/(2\sigma^2)}, \tag{1}$$

where σ is the standard deviation. It then was assumed in [6], [7] that array CGH noise distribution is heavy-tailed. One of heavy-tailed distributions is Student's t [7] in Fig. 2(c) presented by following pdf

$$p(x;\nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}(1+\frac{x^2}{\nu})^{-(\frac{\nu+1}{2})}, \tag{2}$$

where ν is the number of degrees of freedom and Γ is the Gamma function, $\Gamma(z) = \int_0^\infty e^{-t}t^{z-1}dt$, with $z > 0$. Another heavy-tailed distribution in Fig. 2(d) is Cauchy which has the following pdf

$$p(x;\gamma) = \frac{1}{\pi\gamma[1+\frac{x^2}{\gamma^2}]}, \tag{3}$$

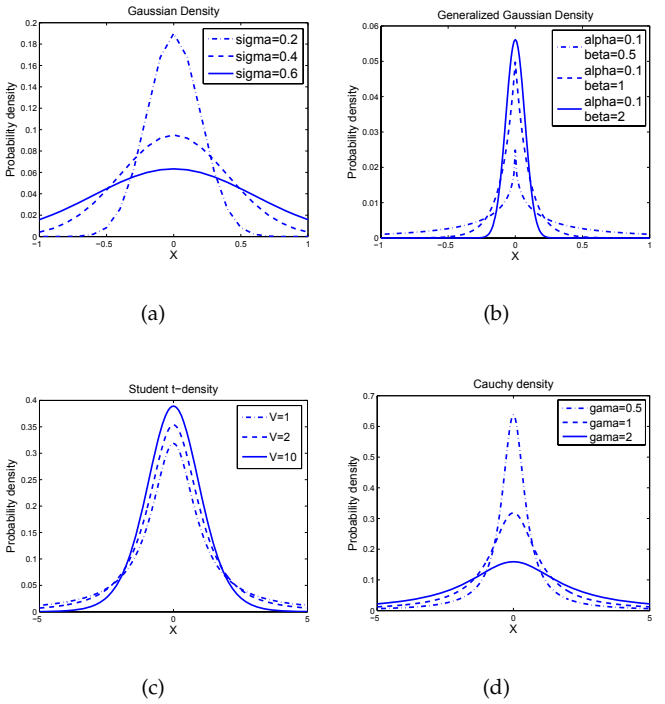


Fig. 2. Four probability density candidates with zero mean: (a) Gaussian density with $\sigma = 0.2, 0.4, 0.6$; (b) Generalized Gaussian density with $\alpha = 0.1, \beta = 0.5, 1, 2$; (c) Student's t density with $\nu = 1, 2, 10$; (d) Cauchy density with $\gamma = 0.5, 1, 2$

where γ is the scale parameter.

Generalized Gaussian distribution (GGD) in Fig. 2(b), which can capture both Gaussian and heavy-tailed, is presented by

$$p(x;\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-(|x|/\alpha)^\beta}, \tag{4}$$

where $\Gamma(.)$ is the Gamma function. Here α is the standard deviation, and β is inversely proportional to the decreasing rate of the peak. α is referred to the scale parameter and β is called as the shape parameter. The Gaussian and Laplacian pdfs are special cases of GGD at $\alpha = 2$ and $\alpha = 1$, respectively. The parameters such as α and β can be estimated as in [21]).

In probability theory and information theory, Kullback-Leibler Divergence (KLD) is the standard way to measure the difference between two probability distributions. If we want to calculate KLD between real probability distribution P and estimated probability distribution Q, we can use the following definition:

$$\Delta H = \sum_i P(i)\log(P(i)/Q(i)). \tag{5}$$

The entropy of distribution P can be calculated by:

$$H(P) = \sum_i P(i)\log(P(i)). \tag{6}$$

We can use $\frac{\Delta H}{H}$ to check how the estimated probability distribution Q fits to real probability distribution P. The

fitting between P and Q is better when $\frac{\Delta H}{H}$ is smaller. We will evaluate the fitting performance of candidates and then propose GGD to approximate noise distribution in array CGH data.

2.3 Validation of New Array CGH Data Noise Model

Four candidates including Gaussian, GGD, Student's t, and Cauchy are employed to fit noise distribution. The real noise of array CGH signal is obtained from eight data sets described above.

To estimate the parameters of Gaussian, Student's t, and Cauchy models, the nonlinear curve-fitting method is used. We calculate $\Delta H/H$ between each model and empirical noise pdf by Eq. (5) and Eq. (6). This $\Delta H/H$ value represents the difference between two distributions devided by the entropy H. A model fits an empirical pdf better than another one when its $\Delta H/H$ is smaller.

Fitting models results of an example are shown in Figs. 3. Histogram of GSM232967's chromosome 15 and fitting results in Fig. 3 illustrate that the difference between GGD model and empirical noise pdf is much less than that of other models. $\frac{\Delta H}{H}$ between GGD model and empirical pdf is 0.0061, while they are 0.0155, 0.0135 and 1.7583 with Gaussian, student's t and Cauchy models, respectively. Another example of fitting models is also shown in supplemental document.

The fitting performance of candidates is evaluated over eight data sets. We compute $\Delta H/H$ between each model and individual empirical pdf of each chromosome from eight data sets and then take the average of these $\Delta H/H$ in each data source as shown in Table 2.

The difference between the GGD model and noise pdf is always the smallest (Table. 2) in all data sources with various platforms. Compared to Gaussian, Student't, and Cauchy models, GGD model is more accurate and sufficient for fitting empirical noise pdf in the array CGH data. Therefore, we propose using GGD model as a new noise model assumption for array CGH data, and will develop a smoothing algorithm based on this GGD noise model. We also notice that both Student's t and GGD models fit the noise distribution better than Gaussian model. Thus the assumption of heavy-tailed noise in array CGH is true and agrees with the conclusion in paper [6].

Data	*Gaussian*	*GGD*	*Student t*	*Cauchy*
Lee 2008	0.0200	**0.0083**	0.0172	0.8846
Snijders 2001	0.0471	**0.0216**	0.0252	0.3154
Bredel 2005	0.0846	**0.0227**	0.0588	0.5770
Smith 2007	0.0298	**0.0184**	0.0259	0.7997
Nicolas 2009	0.0311	**0.0243**	0.0461	0.4238
Kidd 2010	0.0467	**0.0347**	0.0434	1.3228
Perry 2008	0.0183	**0.0144**	0.0156	0.9512
Bovee 2008	0.0304	**0.0166**	0.0228	0.7270

TABLE 2
Average $\Delta H/H$ of four distributions. Samples from eight data sets with various platforms in Table 1 are used for f tting noise models.

3 METHOD

How to reduce heavy-tailed noise and how to detect breakpoints of array CGH data are two central problems in array CGH data processing. In this section, we propose methods to solve for them. First, the generalized Gaussian bivariate shrinkage function based denoising procedure in wavelet domain will be introduced. After that, wavelet derivative scalogram in 1D will be defined to detect breakpoints which mark changing points of segments in array CGH. Finally, we will propose our main method which is a combination of heavy-tailed noise suppression and wavelet pattern-matching for breakpoint detection.

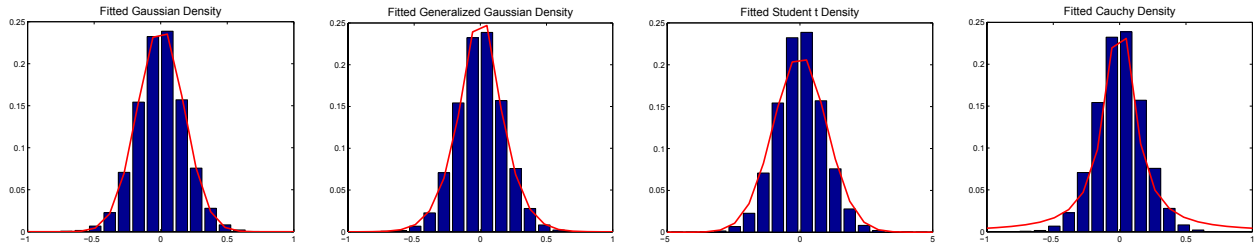
3.1 Heavy-Tailed Noise Suppression

As discussed above, generalized Gaussian is a better noise assumption than Gaussian and the other candidates. With this new noise assumption, denoising becomes a challenging problem. According to empirical comparisons in [5], with Gaussian noise assumption, Wave [9] using stationary wavelet transform (SWT) and hard thresholding showed very good performance. Thus, SWT is still used to reduce noise in this work. However, the hard threshold based estimator is replaced by a new estimator which is designed to operate with heavy-tailed noise.

A simple denoising algorithm via wavelet transform need three steps: decompose the noisy signal by wavelet transform, denoise the noisy wavelet coefficients according to rules, and take the inverse wavelet transform from the denoised coefficients. To estimate wavelet coefficients, the most well-known rules are universal thresholding, soft thresholding [22], [23], [24], and BayesShrink [25]. In these algorithms, the authors assumed that wavelet coefficients are independent. Sendur and Selesnick [26] recently exploited the dependency between coefficients and proposed a non-Gaussian bivariate pdf for the child coefficient w_c and its parent w_p in the complex wavelet transform domain. Huang *et al.* [27] and Nguyen *et al.* [28] applied that function in the complex wavelet transform domain to recover array CGH data with Gaussian noise successfully and got promising results. However, the noise distribution in array CGH that has been proved in previous section is generalized Gaussian rather than Gaussian. Thus, we have to build a new algorithm for new GGD noise model in SWT. Generally we assume that we get the array CGH data Y which includes the deterministic signal D and the independent generalized Gaussian noise $\mathcal{N}$. We have

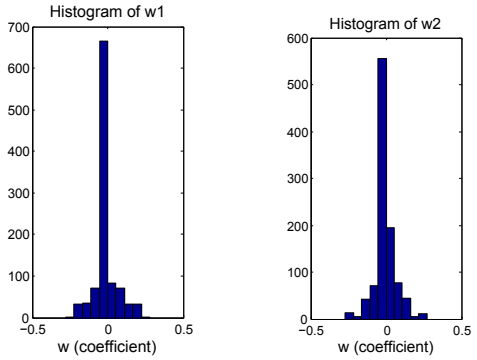
$$Y = D + \mathcal{N}. \tag{7}$$

After decomposing the data Y by the SWT, we get the coefficients $\mathbf{y}_k$ and those coefficients can be formulated



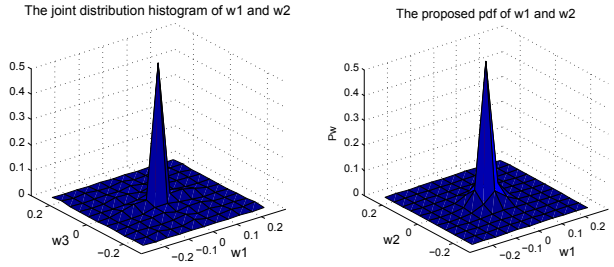
(a) (b) (c) (d)

Fig. 3. The $\Delta H/H$ between the histogram of the chromosome 15 of GSM232967 and four distribution candidates such as (a) Gaussian: $\Delta H/H = 0.0155$, (b) Generalized Gaussian: $\Delta H/H = 0.0061$, (c) student's t: $\Delta H/H = 0.0135$, and (d) Cauchy: $\Delta H/H = 1.7583$.



(a) (b)

Fig. 4. The histograms computed from true array CGH signal. (a) Histogram of w_1 , (b) Histogram of w_2.



(a) (b)

Fig. 5. (a) The joint distribution of w_1 and w_2 created from decomposition of true array CGH signal. (b) The proposed pdf with two variables: w_1 and w_2.

as

$$\mathbf{y_1} = \mathbf{w_1} + \mathbf{n_1}, \quad \mathbf{y_2} = \mathbf{w_2} + \mathbf{n_2}, \tag{8}$$

where $\mathbf{y_1}$ and $\mathbf{y_2}$ are noisy wavelet coefficients, $\mathbf{w_1}$ and $\mathbf{w_2}$ are true coefficients. The joint pdf of noise $\mathbf{n_1}$ and $\mathbf{n_2}$ should follow:

$$p_{\mathbf{n}}(\mathbf{n}) = K(\alpha,\beta)\exp(-\frac{|n_1|^\beta + |n_2|^\beta}{\alpha^\beta}), \tag{9}$$

where $K(\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)}$. The standard MAP estimator [26]) of $\mathbf{w}$ from $\mathbf{y}$ is followed as:

$$\widehat{\mathbf{w}}(\mathbf{y}) = \arg\max_{\mathbf{W}}[\log(p_{\mathbf{n}}(\mathbf{y\text{-}w})) + \log(p_{\mathbf{W}}(\mathbf{w}))]. \tag{10}$$

Fig. 4 illustrates the histogram of $\mathbf{w_1}$ (child) and $\mathbf{w_2}$ (parent). The $\mathbf{w_1}$ and $\mathbf{w_2}$ are computed from array CGH data without noise by using the SWT. Fig. 5 (a) shows the joint distribution of $\mathbf{w_1}$ and $\mathbf{w_2}$. We are going to propose one pdf to fit that joint distribution.

We imitate the idea from [26] and propose a non-Gaussian bivariate pdf for $\mathbf{w_1}$ and $\mathbf{w_2}$ as

$$p_{\mathbf{W}}(\mathbf{w}) = \frac{3}{2\pi\sigma^2}\exp(-\frac{\sqrt{3}}{\sigma}\sqrt{|w_1|^2 + |w_2|^2}). \tag{11}$$

The pdf in Eq. (11) is sketched in Fig. 5 (b). With this pdf, two variables $\mathbf{w_1}$ and $\mathbf{w_2}$ are dependent. Let us define

$$f(\mathbf{w}) = \log(P_w(\mathbf{w})) = \log(\frac{3}{2\pi\sigma^2}) - \frac{\sqrt{3}}{\sigma}\sqrt{|w_1|^2 + |w_2|^2}. \tag{12}$$

Using Eq. (9), Eq. (10) becomes

$$\widehat{\mathbf{w}}(\mathbf{y}) = \arg\max_{\mathbf{W}}[\log(K(\alpha,\beta)) - \frac{|y_1 - w_1|^\beta + |y_2 - w_2|^\beta}{\alpha^\beta} + f(w)]. \tag{13}$$

Solving Eq. (13) is equivalent to solving two following equations

$$sign(y_1 - w_1)\times\beta\times\frac{|y_1 - w_1|^{\beta-1}}{\alpha^\beta} = \frac{\sqrt{3}w_1}{\sigma\sqrt{|w_1|^2 + |w_2|^2}}, \tag{14}$$

$$sign(y_2 - w_2)\times\beta\times\frac{|y_2 - w_2|^{\beta-1}}{\alpha^\beta} = \frac{\sqrt{3}w_2}{\sigma\sqrt{|w_1|^2 + |w_2|^2}}. \tag{15}$$

First, in special case of beta, if this is Gaussian noise ($\beta = 2$ and ${\sigma_n}^2 = \frac{\alpha^2}{2}$), according to [26], the MAP estimator can be formulated as

$$\hat{w_1}(\beta = 2) = \frac{(\sqrt{|y_1|^2 + |y_2|^2} - \frac{\sqrt{3}{\sigma_n}^2}{\sigma})_+}{\sqrt{|y_1|^2 + |y_2|^2}}.y_1, \tag{16}$$

$$\hat{w_2}(\beta = 2) = \frac{(\sqrt{|y_1|^2 + |y_2|^2} - \frac{\sqrt{3}{\sigma_n}^2}{\sigma})_+}{\sqrt{|y_1|^2 + |y_2|^2}}.y_2, \tag{17}$$

where $(u)_+$ is defined by

$$(u)_+ = \begin{cases} 0, & \text{if } u < 0, \\ u, & \text{otherwise.} \end{cases} \tag{18}$$

In Eq. (16) and Eq. (17), σ can be estimated by

$$\hat{\sigma} = \sqrt{(\hat{\sigma}_y^2 - \hat{\sigma}_n^2)_+}, \tag{19}$$

where $\hat{\sigma}_n$ is the noise deviation which is estimated from the finest scale wavelet coefficients using a robust median estimator [23] as follows

$$\hat{\sigma}_n^2 = \frac{median(|y_i|)}{0.6745}, \tag{20}$$

$\hat{\sigma}_y$ is the deviation of observation signal estimated by

$$\hat{\sigma}_y^2 = \frac{1}{M} \sum_{y_i \in N(i)} |y_i|^2, \tag{21}$$

where M is the size of the neighborhood $N(i)$.

Next, in general case of β, we use the successive substitution method to get solution. The parameter estimation procedure is summarized as follows:

Step 1 Initialize $\hat{w_1}^{[0]} = \hat{w_1}(\beta = 2)$ and $\hat{w_2}^{[0]} = \hat{w_2}(\beta = 2)$ at $k = 0$

Step 2 Calculate $r_1{}^{[k]}$ and $r_2{}^{[k]}$ using

$$r_1{}^{[k]} = \sqrt{(\hat{w_1}^{[k]})^2 + (\hat{w_2}^{[k]})^2} |y_1 - \hat{w_1}^{[k]}|^{(\beta-1)} sign(y_1 - \hat{w_1}^{[k]}). \tag{22}$$

$$r_2{}^{[k]} = \sqrt{(\hat{w_1}^{[k]})^2 + (\hat{w_2}^{[k]})^2} |y_2 - \hat{w_2}^{[k]}|^{(\beta-1)} sign(y_2 - \hat{w_2}^{[k]}). \tag{23}$$

Step 3 Find $\hat{w_1}^{[k+1]} = \frac{\beta \sigma r_1{}^{[k]}}{\alpha^\beta \sqrt{3}}$ and $\hat{w_2}^{[k+1]} = \frac{\beta \sigma r_2{}^{[k]}}{\alpha^\beta \sqrt{3}}$

Step 4 Find the differences $\epsilon_1 = \hat{w_1}^{[k+1]} - \hat{w_1}^{[k]}$ and $\epsilon_2 = \hat{w_2}^{[k+1]} - \hat{w_2}^{[k]}$

Step 5 If both $\frac{\epsilon_1}{\hat{w_1}}$ and $\frac{\epsilon_2}{\hat{w_2}}$ are small (less than a threshold), then terminate the iteration. Otherwise, set $k = k + 1$, go to step 2. In theory of successive substitution method, if threshold is less than 10^{-n} with $n > 3$, iteration algorithm can be called convergent. In this work, we choose the threshold of 0.001

3.2 One-Directional Derivative Wavelet Scalogram

After noise suppression step, breakpoints in processed array CGH will be detected by a new 1D scalogram method [29], [30], [31], [32]. True array CGH can be considered as a mixture of unit step functions $h(t)$ ($h = 1\ when\ t >= 0,\ h = 0\ when\ t < 0$) as follows:

$$f(t) = \sum_i^N f_i(t) = \sum_i^N A_i \times h(t - t_{0i}). \tag{24}$$

The continuous wavelet transform can be written as a convolution product in Eq. (25):

$$Wf(u,s) = \int_{-\infty}^{+\infty} f_i(t) \frac{1}{\sqrt{s}} \Psi^\star(\frac{t-u}{s}) dt, \tag{25}$$

where $\star$ is the conjugate, s is scale and u is position in wavelet domain. According to §6 in [33]), the wavelet transform in Eq. (25) can be re-written as a multi-scale differential operator in Eq. (26):

$$W_n f(u,s) = s^n \frac{d^n}{du^n}(f_i * \bar{\theta}_s(t))(u), \tag{26}$$

where $*$ is convolution. In HaarSeg method [10], the simple derivative wavelet (Haar filters) was used. The results of HaarSeg method are promising because of both segmentation results and algorithm speed. However, Haar wavelet is so sensitive to heavy-tailed noise. In this paper, Gaussian wavelet is used instead of Haar wavelet to make sure that our method is robust to noise. Therefore, $\bar{\theta}_s(t)$ can be written as follows:

$$\bar{\theta}_s(t) = \frac{1}{\sqrt{s}} exp(-\frac{t^2}{s^2}). \tag{27}$$

Taking convolution $f_i * \theta_s$, we get result as:

$$Wf(u,s) = A_i \times \int_{t_{0i}}^{+\infty} \frac{1}{\sqrt{s}} e^{\frac{-(t-u)^2}{s^2}} dt. \tag{28}$$

$$W_1 f(u,s) = -A_i \times \sqrt{s} \times e^{\frac{-(u-t_{0i})^2}{s^2}}. \tag{29}$$

$W_1 f(u,s)$ (the first derivative of W) gets maximum at $u = t_{0i}$. The scalogram in 2D is obtained by

$$WS(u,s) = 100 \times \frac{(\frac{W_1 f(u,s)}{\sqrt{s}})^2}{\sum_{i=1}^N (\frac{W_1 f(u,s)}{\sqrt{s}})^2}. \tag{30}$$

However, breakpoint detection using wavelet pattern-matching could not be finished easily in 2D scalogram. So, we change scalogram from 2D to 1D by two following steps. First, ridge lines [33] are identified by linking the local maxima of 2D scalogram at each scale level. L_R and $\mathcal{U}(u)$ represent linking line length and a vector including linked maxima position with u at scale one. In this step, ridge line with length smaller than a certain threshold will be set to zero. The step one can be formulated as

$$\mathcal{U} = \begin{cases} 0, & \text{if } L_R < \text{threshold}, \\ u_1\ u_2\ \cdots u_{s_{max}}, & \text{otherwise.} \end{cases} \tag{31}$$

In the second step, 1D scalogram is built as follows:

$$WS_{1D}(\overline{u}) = \begin{cases} 0, & \text{if } \mathcal{U} = 0, \\ \sum_{u \in \mathcal{U}} WS(u,s), & \text{otherwise,} \end{cases} \tag{32}$$

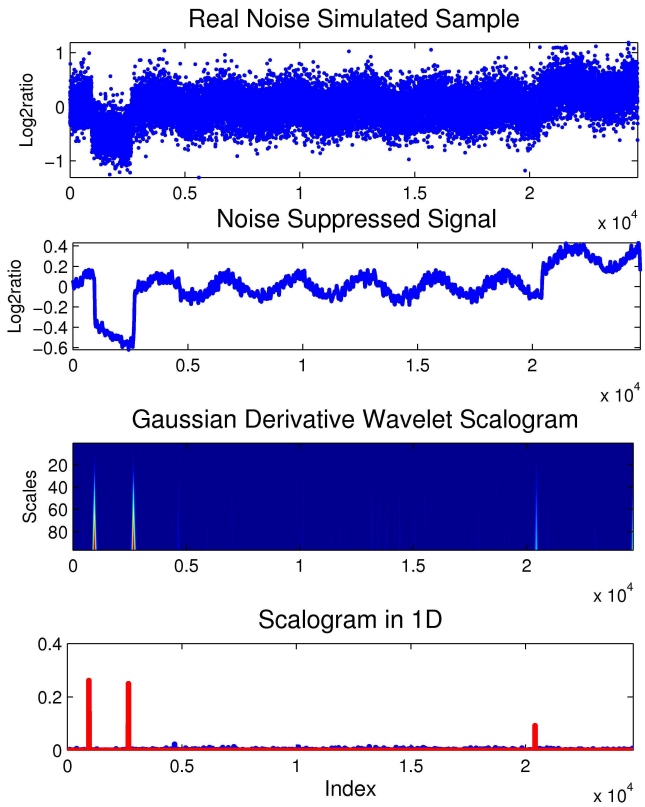
where $\overline{u} = \overline{\mathcal{U}(u)} = \frac{u_1 + u_2 + ... + u_{s_{max}}}{s_{max}}$, s is scale and u is wavelet position.

3.3 Derivative Wavelet Scalogram Based Segmentation (DWSS) Method

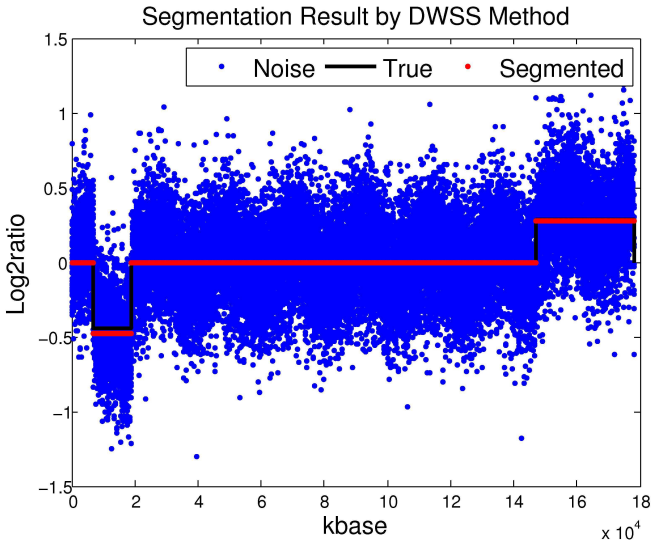
Our new DWSS method is based on two following steps:
Step 1: Noise Suppression. First, heavy-tailed noise in array CGH signal is removed by generalized Gaussian bivariate shrinkage function in stationary wavelet domain. After array CGH is decomposed by SWT, noise suppression will be done by five steps in Section "Heavy-Tailed Noise Suppression". Only high frequency scales are applied to remove noise. Approximation scales are kept to make sure that true signal will not be removed in denoising process. After that, noise suppressed array CGH signal is recovered by inverse SWT.

Step 2: Breakpoint Detection. The 1D Gaussian derivative wavelet scalogram is used to detect breakpoints in noise suppressed array CGH. Mean value of processed signal in each segment will be considered as log2ratio of that segment.

Fig. 6 (a) illustrates two steps of DWSS using a synthetic sample with real noise. First, only outliers with high confidence (high frequency in wavelet domain) are removed. Some noise around true signal is still kept to avoid removing true signal. Because of the simple denoising step, our algorithm can run fast. Gaussian derivative wavelet scalogram in 2D is built from denoised signal. We use 96 scales to build this scalogram. Although footprints of breakpoints are visible in 2D scalogram, breakpoints could not be detected easily. Thus, scalogram in 1D is defined and used to detect breakpoints. The 1D Gaussian derivative wavelet scalogram made our method to be robust with any hybridization bias level of any platform. Signal of hybridization bias can be represented by sin function [8] so it has single frequency which only appears at a few scales (dis-connected line in 2D scalogram). Our method, a wavelet method with 96 scales, removed breakpoints caused by hybridization bias very easily. Only real breakpoints whose frequency bands are very long appear in many scales or all scales (continuous lines 2D scalogram). We demonstrate that 1D scalogram has high values at positions corresponding to breakpoints in array CGH in the fourth sub-figure of Fig. 6 (a). The final segmentation results of DWSS in this example are shown in Fig. 6 (b) in red points. The black line represents true signal (some of them overlap with red color detected segments) and blue points are noisy array CGH data. Our software is available at http://www.nhanguyen.naaan.org/software.html. Evaluations of our method will be discussed in next section.



(a) Demo of two steps in DWSS



(b) Segmentation result by DWSS on one sample

Fig. 6. One example using DWSS method: the sample is generated by adding real noise into a synthetic array CGH (thus known the ground truth of segments: two normals, one deletion and one gain) with bias of 0.2. Noise suppression step and breakpoint detection step are illustrated in (a). DWSS detects exactly four segments (two normals, one deletion and one gain) in this sample in (b).

4 EXPERIMENTS AND DISCUSSIONS

In this section, we first improve array CGH synthetic data model. Next, comparisons between our method and previous works will be performed by using RMSE and ROC curves.

4.1 Improved Synthetic Data Model

Synthetic array CGH data is very important for array CGH study and algorithm evaluation. Because the ground truth of aberration regions is known in synthetic array CGH data, the performances of different smoothing or segmentation algorithms can be measured. However, if the synthetic data model cannot correctly represent the natural properties of real array CGH data, the evaluation results based on them will mislead the array CGH studies. So far, the most commonly used synthetic array CGH data model was proposed by Willenbrock and Fridlyand [4] in 2005. They segmented a primary tumor data set of 145 samples using DNA copy number levels from the empirical distribution of segment

Data Source	α	β
Lee 2008 (40 arrays)	$0.1998 \rightarrow 0.3032$	$1.165 \rightarrow 1.9109$
Smith 2007 (69 arrays)	$0.1221 \rightarrow 0.2010$	$1.0538 \rightarrow 1.7342$
Nicolas 2009 (23 arrays)	$0.2547 \rightarrow 0.3032$	$1.7841 \rightarrow 2.3764$
Kidd 2010 (22 arrays)	$0.04 \rightarrow 0.064$	$1.0793 \rightarrow 1.5167$
Perry 2008 (66 arrays)	$0.043 \rightarrow 0.06$	$1.2037 \rightarrow 1.763$
Bovee 2008 (66 arrays)	$0.049 \rightarrow 0.1$	$1.0296 \rightarrow 2.302$

TABLE 3
Estimated values for the parameters α, β of real array CGH noises

mean values. They got results such as copy number probabilities and the distributions of segment length. The expected $log2ratio$ for each clone was computed as $\log_2(\frac{cP_t+2(1-P_t)}{2})$ where c is the assigned copy number with P_t is a proportion of tumor cells whose values are from a uniform distribution between 0.3 and 0.7.

Following this standard model, we create true array CGH signal without noise. In order to improve the model [4], we first add the probe hybridization bias to true signal that was proposed in [8]. The simulated signal can be written as:

$$Y = D + \mathcal{R} + \mathcal{N}, \tag{33}$$

where D is the true signal, $\mathcal{R}$ is the hybridization bias and $\mathcal{N}$ is the noise. We use a parameter b whose value is from zero to one to adjust bias value in simulated data as follows

$$\mathcal{R} = b \times (0.5\sin(2\pi 0.001m) + Gaussian(0, 0.25)), \tag{34}$$

where m is the length of simulated signal and b is the bias value. The old model in [4], Gaussian noise is added to the true signal. In this work, we introduce two kinds of synthetic data by adding real noise or GGD noise into true signal.

4.1.1 GGD Noise:

As discussed in previous section, GGD fits noise pdf in array CGH data very well. Parameters α and β are estimated as shown in Table 3. From Table 3, we observe that the parameter α ranges from 0.12 to 0.3 and the parameter β ranges from 1.05 to 2.38. Therefore, GGD noise model with α and β values in Table. 3 will be used for synthetic array CGH data generation.

4.1.2 Real Noise:

We extract the real noise from array CGH data by following steps. First, we calculate the histogram of DNA copy numbers of a chromosome from the real self-self test array CGH data as shown in Fig. 7 (a), *i.e.* the noise histogram. From this histogram, a discrete pdf with 64 bins is formed as Fig. 7 (b). Then we interpolate the 64 bins-pdf and normalize to get a new pdf with some thousands of bins as Fig. 7 (c). Finally a new random noise vector will be created from this pdf. In this experiment, we use 286 arrays with ten thousands probes from various data sets as shown in Table 3, which contain noise only. Therefore, we have 286 pdfs to create thousands of random noise vectors which are added to true signal to create real noise based simulated array CGH data. The whole procedure to extract real noise is illustrated in Fig. 7.

After adding noise, we create unequally spaced probes suggested by [3]. The intuition of this step is that the distances between probe k and probe $k+1$ are randomly and the best way to get these distances from the real array CGH data, such as Lee 2008 array [12] for high resolution data. We then place unequally spaced probes on chromosomes. The number of probes can be low, high and very high. Now, we generate many artificial chromosomes of length 200 Mbase with three resolutions and two kinds of noise including generalized Gaussian noise and real noise.

4.2 Performance Evaluations of DWSS Method

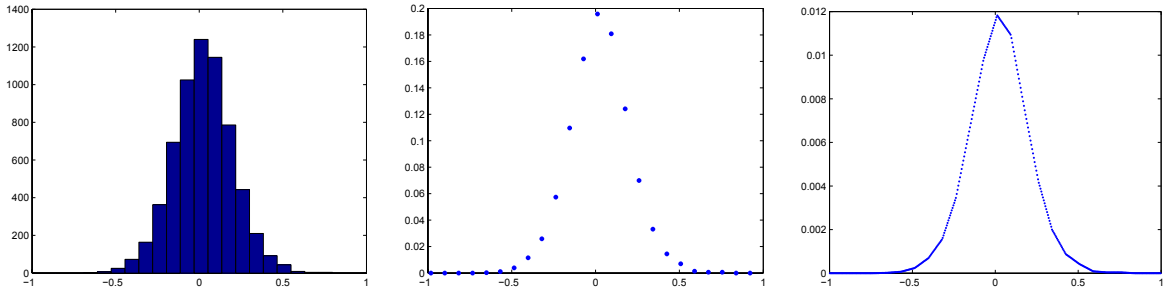
We compare our DWSS method to other state-of-the-art methods, including Lowess [5], Wave [9], Smoothseg [7], HaarSeg [10], CBS [2], GADA1 [11], and GADA2 [8]. We used R package smoothseg for smoothseg, and waveslim for Wave. With Lowess, HaarSeg, GADA1, GADA2, CBS and our method, MATLAB implementations are used. HaarSeg, GADA1, and GADA2's implementations are downloaded from sharing links in [8], [10].

4.2.1 RMSE Comparisons:

For each of three noise models (Gaussian, GGD with heavy-tail, and real noise), we create one thousand high resolution chromosomes. For every 1000 generated chromosomes, bias of 0.2, 0.4 and 0.8 are applied respectively to obtain 3000 chromosomes. After applying each method to the simulated data, we use the root mean square errors (RMSEs) to measure the differences between ground truth and segmentation results of all methods. All averaged results are shown in Fig. 8 (a) and the DWSS method has the best performance. The DWSS outperforms averagely the Lowess by 78%, the Wave by 77%, the Smoothseg by 82%, the HaarSeg by 51%, the CBS by 78%, the GADA1 by 78%, and the GADA2 by 40% in terms of the RMSEs. For all noise models, the DWSS consistently achieves much better results than the others.

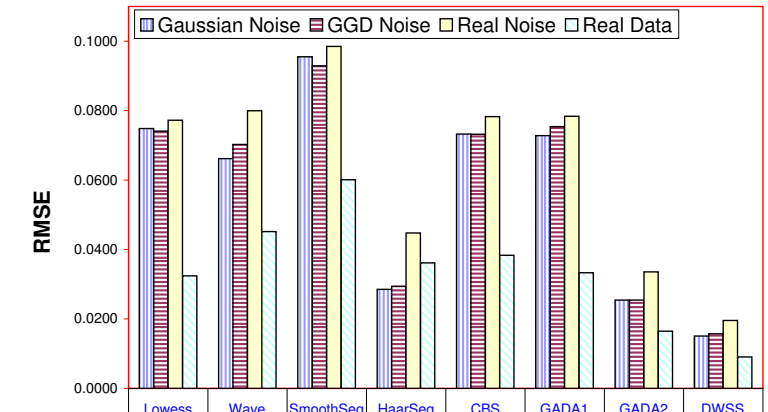
Real array CGH data is also used to evaluate performances of eight above methods. Lee 2008 array [12] array including 40 samples, Smith2007 array including 69 samples and Nicolas 2009 including 23 samples are three real data with the known ground truth. In Fig. 8 (a), the performance of our DWSS method is much better than that of others. Average RSME of our method is smaller than SmoothSeg by 5.7 times, Lowess, CBS and GADA1 by 4.5 times, Wave by 4.3 times, HaarSeg by 2 times and GADA2 by 1.7 times.

In general, if we consider both simulated and real data, our method improved previous methods 41% to 77%.



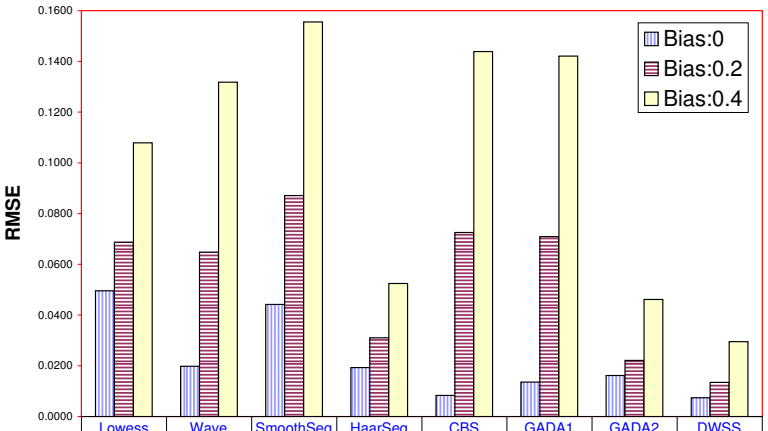
(a) (b) (c)

Fig. 7. The procedure to create real noise from chromosome 19 of GSM232967. (a) Histogram with 64 bins, (b) pdf of 64 bins, (c) pdf of 1024 bins.



	Lowess	Wave	SmoothSeg	HaarSeg	CBS	GADA1	GADA2	DWSS
Gaussian Noise	0.0749	0.0662	0.0955	0.0285	0.0733	0.0728	0.0254	0.0151
GGD Noise	0.0741	0.0703	0.0929	0.0294	0.0732	0.0754	0.0254	0.0157
Real Noise	0.0772	0.0800	0.0985	0.0448	0.0783	0.0784	0.0336	0.0196
Real Data	0.0324	0.0452	0.0601	0.0361	0.0383	0.0333	0.0165	0.0091

(a) Different noise sources



	Lowess	Wave	SmoothSeg	HaarSeg	CBS	GADA1	GADA2	DWSS
Bias:0	0.0496	0.0198	0.0442	0.0193	0.0083	0.0136	0.0162	0.0074
Bias:0.2	0.0687	0.0648	0.0871	0.0310	0.0726	0.0709	0.0221	0.0134
Bias:0.4	0.1079	0.1318	0.1556	0.0524	0.1439	0.1421	0.0461	0.0295

(b) Different bias levels

Fig. 8. Average RMSEs of all testing methods on three simulated data sets and real data are shown. We test different bias levels with each simulated data set and take average RMSE. We use three real data sets and also take average RMSE.

4.2.2 ROC Curve Comparisons:

A comparison of array CGH detection algorithms was studied by [5]. They used the ROC curve to evaluate 11 algorithms with aberration widths of 5, 10, 20 and 40, and signal-to-noise ratios (SNRs) of 1, 2, 3 and 4. Many synthetic chromosomes consisting of 100 probes are created from four templates with Gaussian noise and square-wave signal at the center of chromosome. In 2007, Huang *et al.* [7] improved this setting by decreasing the width of the center square-wave and increasing the noise level. In [10], Ben and Eldar proposed using very high resolution data and real noise to improve the quality of evaluation. In this paper, we evaluate the performance of all methods not only at the middle of signal but also at the border of signal. Therefore, we use four templates with the aberrations at the center and four more templates with the aberrations at the border. The aberration widths used in this paper are 5%, 10%, 15% and 20% of whole chromosome length. Both Gaussian and real noise are used to evaluate all methods. The real noise from forty self-self test arrays of Lee 2008 array [12] is also add to these templates. In all cases, bias of 0.8 will be added to make problem harder. Using all eight genomic templates, 100 noisy arrays are generated with unequally spaced probes. We test segmentation performance of all methods on three true segment amplitudes of $log_2\frac{3}{2}$, $log_2\frac{4}{2}$ and $log_2\frac{5}{2}$. The ROC curves of eight methods using four different data are plotted in Fig. 9.

In Fig. 9 (d), when real noise simulated data which has gain segmentation amplitude of $log_2\frac{5}{2}$ is used, the performances of DWSS and GADA2 are the best. HaarSeg and CBS work well and their ROC curves are very close to each other. Wave method also works well. The next ones are GADA1, SmoothSeg, and Lowess. The gain segment amplitude is further reduced to $\log_2\frac{3}{2}$. With copy of three in Figs. 9 (a)(c) and (b), all methods get worse results. However, DWSS is still the best one. The next ones are HaarSeg and GADA2. These results are consistent with the above results using RMSEs as evaluation metric.

By using both RMSEs and ROC curves, we can conclude that DWSS has the best performance. GADA2 and HaarSeg are good methods being robust with bias. GADA2 is more robust with heavy-tailed than HaarSeg. HaarSeg detects segments which have small signal-noise ratio better than GADA2. GADA1 and CBS are comparable. About algorithm speed, DWSS runs faster than CBS by 2.96 times (Table 4 in supplemental doc).

4.3 DISCUSSIONS

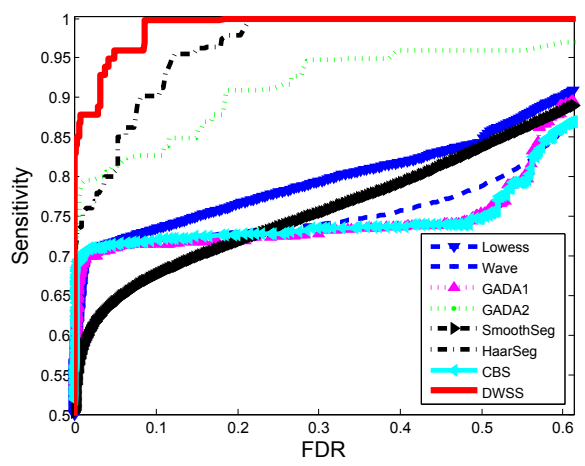
Based on the experimental results in Fig. 9, we further investigate all eight methods and also list their results by different bias levels in Fig. 8 (b). Lowess method [5] is robust to heavy-tail, but very sensitive to bias. In general, the performance of Lowess is not good. Wave method [9] should be used with Gaussian noise and without bias, because it was only designed for Gaussian noise. With small SNR, the performance of Wave method is comparable with CBS and GADA1 methods. SmoothSeg method [7] is designed for heavy-tailed noise, hence it operates well with generalized Gaussian noise. However, compared to other methods, SmoothSeg method has a worse performance. HaarSeg method [10] uses wavelet based pattern matching so that it is robust to bias. However, since Haar filter is used in stationary wavelet domain, it is sensitive to outlier or it is not robust to heavy-tailed noise. Overall HaarSeg still gives promising results. It is better than denoising method and two segmentation methods such as CBS and GADA1. If compared by ROC curves, HaarSeg works even better than GADA2 in case of small SNR.

CBS method [2] is the second best in case without bias. That means CBS is robust to heavy-tailed noise. However, it is very sensitive to bias. Thus, it gets much worse results if signal has bias. GADA1 method [11] is comparable with CBS and is less robust to heavy-tailed noise than CBS. This method can run much faster than CBS. It also has problem with bias as CBS. GADA2 method [8] was designed to operate with probe hybridization bias. It is also robust to heavy-tailed noise. This method is better than GADA1, CBS, HaarSeg and all denoising methods. However, because it was designed to be robust with hybridization bias, this method loses some segments with small energy. With small SNR segments, GADA2 works worse than DWSS and HaarSeg. Except for this disadvantage, GADA2 was the best method so far.

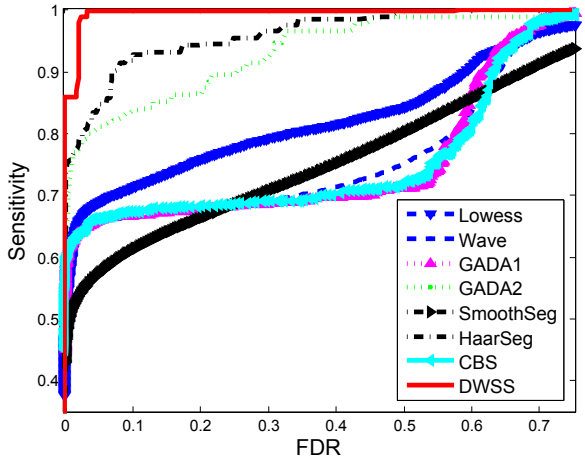
In general, Lowess, Wave, SmoothSeg, GADA1 and CBS methods are sensitive to bias. Our DWSS method always shows the best performance in all different conditions. GADA2 and HaarSeg have the second best performance.

5 CONCLUSION

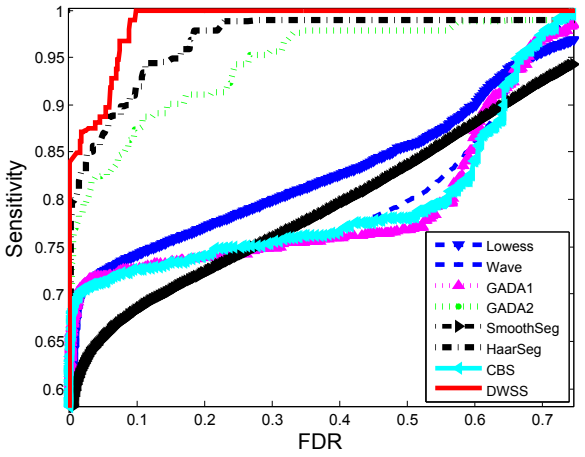
In this paper, we examined noise distribution in array CGH data using eight real data sets in many platforms with different resolutions. When compared with other distributions used in previous research such as Gaussian and Student's t distributions, the generalized Gaussian distribution fits very well noise pdf in array CGH data. Therefore we proposed using GGD for modeling noise distribution in the array CGH data and developed a novel smoothing-segmentation method based on this generalized Gaussian noise. Bivariate shrinkage function's theory in SWT is built with an approach to suppress heavy-tailed noise in array CGH. One-directional Gaussian wavelet derivative scalogram is defined and proposed to detect breakpoints in array CGH. Because the ground truth aberration regions are not clear in real array CGH data sets, synthetic array CGH data plays an important role in array CGH analysis algorithm evaluation. By using generalized Gaussian noise and real noise, we also improved the synthetic array CGH data models which are closer to the real array CGH data than the most commonly used standard [4] and [8]. Both synthetic data and real data are used to evaluate the



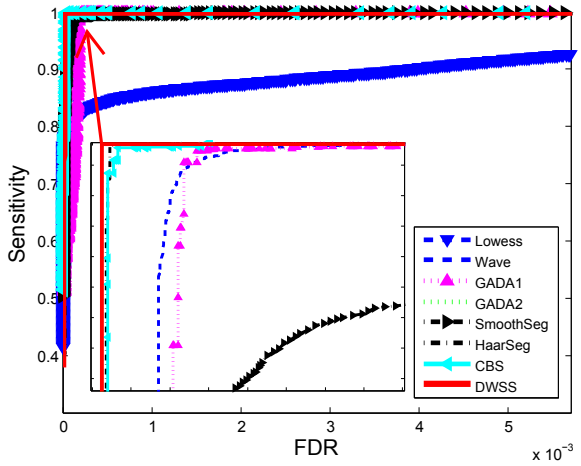
(a) Gaussian noise and segment gain of $\log_2 \frac{3}{2}$



(b) GG noise and segment gain of $\log_2 \frac{3}{2}$



(c) Real noise and segment gain of $\log_2 \frac{3}{2}$



(d) Real noise and segment gain of $\log_2 \frac{5}{2}$

Fig. 9. Results on one hundred simulated samples with true abnormal segments amplitude of $\log_2 \frac{3}{2}$ and $\log_2 \frac{5}{2}$, respectively. Bias of 0.8 is used. ROC curves are obtained from arrays which are generated from 8 genomic templates and both Gaussian and real noise sources. In four different conditions, DWSS gives the best performance. The second one is HaarSeg.

performance of our method, DWSS. We demonstrated our new method outperforms other most commonly used algorithms in array CGH literature both in terms of RMSE and ROC curve.

REFERENCES

[1] P. Eilers and R. de Menezes, "Quantile smoothing of array CGH data," *Bioinformatics*, vol. 21, pp. 1146–1153, 2005.

[2] A. Olshen, E. Venkatraman, R. Lucito, and M. Wigler, "Circular binary segmentation for the analysis of array-based DNA copy number data," *Biostatistics*, vol. 5, pp. 557–572, 2004.

[3] Y. Wang and S. Wang, "A novel stationary wavelet denoising algorithm for array-based DNA copy number data," *International Journal of Bioinformatics Research and Applications*, vol. 3, no. 2, pp. 206 – 222, 2007.

[4] H. Willenbrock and J. Fridlyand, "A comparison study: applying segmentation to array CGH data for downstream analyses," *Bioinformatics*, vol. 21, no. 22, pp. 4084–4091, 2005.

[5] W. Lai, M. Johnson, R. Kucherlapati, and P. Park, "Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data," *Bioinformatics*, vol. 21, pp. 3763–3770, 2005.

[6] J. Hu and *et al.*, "Exploiting noise in array CGH data to improve detection of DNA copy number change," *Nucleic Acids Research*, vol. 35, 2007.

[7] J. Huang and *et al.*, "Robust smooth segmentation approach for array CGH data analysis," *Bioinformatics*, vol. 23, pp. 2463–2469, 2007.

[8] R. Pique-Regi, A. Ortega, and S. Asgharzadeh, "Joint estimation of copy number variation and reference intensities on multiple dna arrays using gada," *Bioinformatics*, 2009.

[9] L.Hsu, SG.Self, D.Grove, T.Randolph, K.Wang, JJ.Delrow, L.Loo, and P.Porter, "Denoising array-based comparative genomic hybridization data using wavelets," *Biostatistics(Oxford,England)*, vol. 6, no. 2, pp. 211–226, 2005.

[10] E. Ben and Y. Eldar, "A fast and flexible method for the segmentation of aCGH data." *Bioinformatics*, vol. 24, pp. 139–145, 2008.

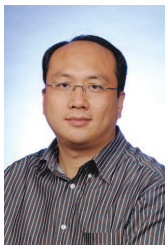
[11] R. Pique-Regi, J. Monso-Varona, A. Ortega, R. Seeger, T. Triche, and S. Asgharzadeh, "Sparse representation and bayesian de-

tection of the genome copy number alterations from microarray data," *Bioinformatics*, 2008.
[12] A. Lee and *et al.*, "Analysis of copy number variation in the rhesus macaque genome identifies candidate loci for evolutionary and human disease studies." *Hum Mol Genet*, vol. 17, no. 8, pp. 1127–36, 2008.
[13] A. Snijders and *et al.*, "Assembly of microarrays for genome-wide measurement of dna copy number," *Nature Genetics*, vol. 29, pp. 263 – 264, 2001.
[14] M. Bredel and *et al.*, "High-resolution genome-wide mapping of genetic alterations in human glial brain tumors," *Cancer Research*, vol. 65, pp. 4088–4096, 2005.
[15] A. Smith and *et al.*, "Array CGH analysis of copy number variation identifies 1284 new genes variant in healthy white males: implications for association studies of complex diseases." *Human Mol. Genet.*, 2007.
[16] T. Nicholas, Z. Cheng, M. Ventura, K. Mealey, E. Eichler, and J. Akey, "The genomic architecture of segmental duplications and associated copy number variants in dogs," *Genome Res.*, 2009.
[17] J. Kidd, N. Sampas, F. Antonacci, T. Graves *et al.*, "Characterization of missing human genome sequences and copy-number polymorphic insertions," *Nat Methods*, vol. 7, pp. 365–71, 2010.
[18] G. Perry, A. Ben-Dor, A. Tsalenko, N. Sampas *et al.*, "The fine-scale and complex architecture of human copy-number variation," *Am J Hum Genet*, vol. 82, pp. 685–95, 2008.
[19] D. Bovee, Y. Zhou, E. Haugen, Z. Wu *et al.*, "Closing gaps in the human genome with fosmid resources generated from multiple individuals," *Nat Genet*, vol. 40, pp. 96–101, 2008.
[20] N. Nguyen, H. Huang, S. Oraintara, and Y. Wang, "Denoising of array-based DNA copy number data using the dual-tree complex wavelet transform," *IEEE BIBE07*.
[21] M. Do and M. Vetterli, "Wavelet-based texture retrieval using generalized Gaussian density and kullback-leibler distance," *IEEE Transactions on Image Processing*, vol. 11, pp. 146–158, 2002.
[22] D. Donoho and I. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.
[23] D. Donoho, "De-Noising by soft-thresholding," *IEEE Trans. on Inf. Theory*, vol. 41, no. 3, pp. 613–627, 1995.
[24] I. Johnstone and B. Silverman, "Wavelet threshold estimators for data with correlated noise," *Journal of the Royal Statistical Society*, no. 59, pp. 319–351, 1997.
[25] S. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans.Image processing*, vol. 9, pp. 1532–1546, Sept.2000.
[26] L. Sendur and I. Selesnick, "Bivariate shinkage function for wavelet-based denoising exploiting interscale dependency," *IEEE Transaction on Signal Processing*, vol. 50, no. 11, November 2002.
[27] H. Huang, N. Nguyen, S. Oraintara, and A. Vo, "Array CGH data modeling and smoothing in stationary wavelet packet transform domain," *BMC Genomics*, vol. 9, p. S2:S17, 2008.
[28] N. Nguyen, H. Huang, S. Oraintara, and A. Vo, "Stationary wavelet packet transform and dependent laplacian bivariate shrinkage estimator for array-cgh data smoothing," *Journal of Computational Biology*, 2010.
[29] N. Nguyen, A. Vo, and K. Won, "A wavelet-based method to exploit epigenomic language in the regulatory region," *Bioinformatics*, vol. 30, pp. 908–914, 2014.
[30] ——, "A wavelet approach to detect enriched regions and explore epigenomic landscapes," *Journal of Computational Biology*, vol. 21, pp. 846–854, 2014.
[31] N. Nguyen and K. Won, "Gaussian derivative wavelets identify dynamic changes in histone modification," CIBCB. IEEE, May 2014.
[32] N. Nguyen, A. Vo, and K. Won, "A stationary wavelet entropy based clustering approach accurately predicts gene expression," *Journal of Computational Biology*, vol. 22, pp. 236–249, 2015.
[33] S. Mallat, *Wavelet Tour of Signal Processing - The Sparse Way*. Elsevier, 2009.

Nha Nguyen received his B.S and M.S degrees in Electrical Engineering from HCMC University of Technology, Viet Nam, in 1996 and 2000, respectively. He worked in the Sai Gon Technology University, Viet Nam, as a lecturer in the Department of Electrical Engineering from 2001 to 2007. He received the Ph.D. degree in electrical engineering from the University of Texas at Arlington in 2010. He joined the Perelman School of Medicine, University of Pennsylvania, PA, as a Postdoctoral Research Fellow in 2011, and became a Research Scientist in 2016. His research interests include signal processing in bioinformatics and biomedical image analysis.

An Vo (S'07-M'11) received the B.S. and M.S. degrees in electrical engineering from HCMC University of Technology in 1997 and 2000, respectively, and the Ph.D. degree in electrical engineering from the University of Texas at Arlington in 2008. From 2000 to 2004, she was a Lecturer in the Department of Electrical Engineering at the HCMC University of Technology. She joined the Feinstein Institute for Medical Research, Northwell Health, NY, as a Postdoctoral Research Fellow in 2009, and became a Research Scientist and an Assistant Investigator in 2011 and 2014, respectively. In July 2014, she was also appointed Assistant Professor with the Department of Molecular Medicine Research, at the Hofstra-Northwell School of Medicine, Hofstra University, NY. Her current research interests include investigation and development of new algorithms and mathematical tools for the advanced processing of biomedical images and biological signals to improve the diagnosis and understanding of diseases and complex biological systems.

Haibin Sun received his BS degree in Mechanics from Shandong Jianzhu University in 1997. He received his MS degree in mechatronic engineering and PhD degree in Computer Science from Jilin University in 2002 and 2006, respectively. He visited the University of Texas at Arlington, USA as a visiting scholar in 2013. He has published many academic papers in conferences and respectable journals such as lecture notes in computer science, knowledge-based systems, etc . His research interests include artif cial intelligence, spatial reasoning, machine learning and data mining. He is currently an associate professor at Shandong University of Science and Technology. Other interests include embedded system, wireless sensor network.



Heng Huang received both B.S. and M.S. degrees from Shanghai Jiao Tong University, Shanghai, China, in 1997 and 2001, respectively. He received the Ph.D. degree in Computer Science from Dartmouth College in 2006. He started working as an assistant professor in Computer Science and Engineering Department at University of Texas at Arlington in 2007, and became a tenured associate professor at the same department in 2013. Since 2015, he has been a full professor at the same department. He became the Distinguished University Professor at 2017. His research interests include machine learning, data mining, bioinformatics, neuroinformatics, and health informatics.