Published online 15 June 2010

Nucleic Acids Research, 2010, Vol. 38, No. 15 ***e157***
doi:10.1093/nar/gkq548

FACADE: a fast and sensitive algorithm for the segmentation and calling of high resolution array CGH data

Bradley P. Coe*, Raj Chari, Calum MacAulay and Wan L. Lam

Department of Integrative Oncology, British Columbia Cancer Research Centre, Vancouver, B.C., Canada

Received December 28, 2009; Revised May 26, 2010; Accepted May 30, 2010

ABSTRACT

The availability of high resolution array comparative genomic hybridization (CGH) platforms has led to increasing complexities in data analysis. Specifically, defining contiguous regions of alterations or segmentation can be computationally intensive and popular algorithms can take hours to days for the processing of arrays comprised of hundreds of thousands to millions of elements. Additionally, tumors tend to demonstrate subtle copy number alterations due to heterogeneity, ploidy and hybridization effects. Thus, there is a need for fast, sensitive array CGH segmentation and alteration calling algorithms. Here, we describe Fast Algorithm for Calling After Detection of Edges (*FACADE*), a highly sensitive and easy to use algorithm designed to rapidly segment and call high resolution array data.

INTRODUCTION

Array comparative genomic hybridization (CGH) is a high-resolution tool for the identification of regions of copy number gain and loss in normal and disease genomes (1,2). Genome profiling efforts with array CGH have led to many significant discoveries ranging from identification of oncogenes and tumor suppressors in cancer, to genes associated with mental retardation syndromes, and identification of genetic variation between healthy individuals (2–6). As modern array CGH platforms have increased in density to new platforms containing hundreds of thousands to millions of data points, complexity in data analysis has likewise increase (1).

Two critical processes must be applied to normalized array CGH data in order to allow accurate downstream analysis: segmentation and calling. Segmentation involves the identification of putative breakpoints in the data that delineate regions of equal copy number, thereby defining a contiguous region of an alteration. On its own, segmented data is useful for identification of frequent breakpoints and applications where the absolute ratio of test to reference DNA is important. However this is complicated by the fact that array CGH only detects segmental changes in copy number, not overall ploidy, and moreover, many samples represent a mix of normal and disease cells (1,7). These factors combined with biases in hybridization, lead to observed signal ratios that are significantly attenuated in comparison to the ideal 3:2 or 1:2 ratios for single copy gains and losses respectively (1,7,8). Thus, it is highly beneficial to convert array data to a called format. In calling, segmented regions are annotated as being gained or lost through conversion of the continuously distributed raw copy number ratios to a representation of the underlying discrete distribution of copy number states (0–4 or more copies) that is actually being measured by array CGH. This data thus best represents the DNA copy number being measured and is arguably the most straightforward application for downstream analysis (8).

Experimental limitations such as a weak signal in a background of noise are not the only factors that complicate array analysis. With the emergence of ultra high density arrays, computational analysis is a growing challenge. The high data content of modern array platforms can lead to impracticable execution times for current algorithms. Many current algorithms were designed/tested on low-resolution platforms (such as 1 Mb resolution BAC arrays and cDNA platforms), for which exists a plethora of public data to allow verification (9,10). Thus, their utility on newer arrays is not necessarily tested and execution times are often not optimized with such high-density data in mind. This has resulted in many current algorithms, which despite their high accuracy, may take hours to months to process large sample sets of high-resolution data. Some recent algorithms can take 20 h just to process a single chromosome on modern hardware, or demonstrate an exponential relationship between processing time and array density complexity

*To whom correspondence should be addressed. Tel: +11 604 675 8111; Fax: +11 604 675 8232; Email: bcoe@bccrc.ca

precluding application to high resolution data sets (11,12). This computational complexity is simply not sustainable in most molecular biology laboratories, which are becoming increasingly reliant on the analysis of large-scale genomic array data.

Most of the fast algorithms that have been developed to date offer only segmentation/smoothing without statistical inference (13–15). Additionally, although fast algorithms exist for CNV (copy number variation) detection in high resolution SNP arrays, these algorithms are not applicable to complex genomes such as those observed in cancer specimens, and are often based on a threshold based analysis of segmentation only (16). Another effective approach to speeding up execution times is the development of algorithms that run in parallel, or cluster environments with multiple CPUs. However this costly approach may not be a practical solution for experimental biology laboratories (17).

This combination of weak signal in conjunction with high-density data demands a highly sensitive, rapid algorithm for efficient data analysis. In this study, we present Fast Algorithm for Calling After Detection of Edges (*FACADE*). *FACADE* represents a highly accurate Java based application for the segmentation and calling of array CGH data that demonstrates fast execution times, and a near linear relationship between execution time and array density to allow application to both current and future platforms.

MATERIALS AND METHODS

Edge detection

FACADE takes as its primary input, a matrix of $\log_2$ ratios ($\log_2$ ratios represent a comparison of normalized signal intensities from hybridization of labeled sample and reference genomes) for array elements spanning the genome. The first step in the *FACADE* algorithm is to calculate an underlying baseline distribution X, defined as a spatially uniform distribution of $\log_2$ ratios generated from the array elements at spacing defined by the user. For tiling BAC arrays this is calculated by determining the average $\log_2$ ratio value (average of overlapping clones at a single point) at the user specified increments (optimally 5 kb). For higher density arrays which have >75 000 data points, a moving average is calculated based on a user supplied window size and increment. The default value for window size is 5 kb, which is applicable to most high resolution arrays, while the recommended default increment for high resolution arrays is 2 kb. The aim of this step is to apply minimal smoothing to the data while converting the variable array element spacing to a uniform set of data points. Smoothing of noise is accomplished through the first and second derivative edge detection. When the baseline hits a gap in the array element distribution <1.5 Mb in size the baseline values are defined based on a linear fit between the array elements on either boundary of the gap. Larger gaps are annotated and automatically assigned as a potential breakpoint.

Next the first derivative X' of the $\log_2$ ratio is calculated for all positions X_i (where i is genomic position along each chromosome) using a smoothing kernel based on the first derivative of a gaussian with σ defined by the user (18):

$$X'_i = \sum_{j=-(3\sigma+1)}^{j=3\sigma+1} \frac{j}{\sigma^3\sqrt{2\pi}} e^{-j^2/2\sigma^3} \cdot X_{i+j} \quad (1)$$

The second derivative, X''_i is then calculated for all positions X_i using a smoothing kernel based on the Difference of Gaussians (DoG) operator (19):

$$X''_i = \sum_{j=-(3\sigma+1)}^{j=3\sigma+1} \left(\frac{1}{\sigma\sqrt{2\pi}} e^{-j^2/2\sigma^2} - \frac{1}{(\sigma/1.6)\sqrt{2\pi}} e^{-j^2/2(\sigma^2/1.6)} \right) \cdot X_{i+j} \quad (2)$$

For Equations (1) and (2), when X_{i+j} points to an index beyond the chromosome maximum of i (i_{chrmax}), the data is mirrored by replacing the value of $i+j$ with $i_{chmrax} - [(i+j) - (i_{chrmax}+1)]$.

Local peaks (positive and negative) in X' represent potential breakpoints (18 and Supplementary Figure S1), and are identified through application of an iterative thresholding procedure to $|X'|$ until the number of peaks identified converges with the number of breakpoints parameter specified by the user ± 10%. The X'' values are then examined for all potential breakpoint regions (X' peaks) and zero crossings (change in sign of X'' at a peak in X' or edge in X) are used to define the position of the breakpoint within the peak [(19) and Supplementary Figure S1). This two-step process helps to eliminate the overcalling of false edges that could occur when utilizing only a DoG operator on noisy data (Supplementary Figure S1). It should be noted that the number of breakpoints parameter is meant to be an overestimate of the number of breakpoints detected, as the calling procedure will remove false segments and merge adjoining segments of the same copy number. Thus, it is only critical that the parameter is set to a sufficiently high level to catch all true breakpoints.

Iterative search for copy neutral control regions

In order to determine if a segment represents copy number gain, loss or no change, contiguous stretches of the genome which are likely to exhibit normal copy number, are identified to function as control regions. This is performed by an iterative search procedure (independent of segmentation), which identifies a 10 element (or 20 element for high density arrays) consecutive stretch on each chromosome where the absolute value of the sum of the average and standard deviation of the $\log_2$ ratios is minimized. We then select the five regions from across the genome with the minimal average and standard deviation and combine them to generate a set of copy number neutral elements for statistical comparisons.

Alteration calling

Next, *FACADE* segments the genome based on the putative breakpoints. The number of segments constructed on each chromosome is equal to the number of

breakpoints +1. For each segment, we determine the number of array elements within the segment and the mean $\log_2$ ratio of those elements.

Next, we assign states to each segment, using either 3 or 5 (default) unique states depending on a user parameter selected in the application: '5 Level Output' or '3 Level Output'. The 5 Level Output option identifies high level amplification events and homozygous deletions distinctly from single copy gains and losses. The 3 Level Output option is for use in cases where only gain, normal, and loss calls are needed. If the 5 level option is selected, the states are (−2, −1, 0, +1, +2) representing 0, 1, 2, 3 and 4+ copies (in a diploid genome) respectively. Alternatively, if the 3 level option is selected, the states are (−1, 0, +1), representing <2 copies, 2 copies, and >2 copies respectively. This is accomplished through the following procedure:

Prior to statistical evaluation, we apply two filtering criteria to establish −2, 0 and +2 states. First, segments whose mean $\log_2$ ratio is greater than the user supplied amplification threshold or less than the deletion threshold are assigned the amplification (+2) and deletion (−2) states without further testing. This step is only performed if 5 level mode is selected. Secondly, segments for which the absolute value of the $\log_2$ ratio mean is less than the user supplied $\log_2$ ratio delta parameter are assigned to the normal (0) copy number state without further statistical examination.

The remaining segments represent putative copy number alterations and are tested by statistical comparison with the set of copy number neutral control elements by using the Mann–Whitney U-test. Due to the sensitivity of this test, long segments with consistent but low level ratio shifts may generate false positive calls. To account for this we apply an empirically derived correction whereby the *P*-value is multiplied by the length of the segment (in array elements). Segments whose *P*-value is less than the *P*-value threshold defined by the user are assigned to a copy number state of gain (+1) if the segment $\log_2$ ratio mean is >0 and to a copy number state of loss (−1) if the segment $\log_2$ ratio mean is <0. Segments with *P*-values greater than or equal to the *P*-value threshold are assigned a copy number state of normal (0).

Due to the nature of the edge detection parameters, and increased noise in regions of gain and loss, it is common for two adjacent segments to be of the same copy number. Adjacent segments are thus merged if both segments demonstrate the same call value. Additionally segments which are separated by up to two data points with a copy number state of normal (0) are joined. After merging, the *P*-value and $\log_2$ ratio mean for the new segment are recalculated.

In cases where the σ specified by the user is >5, it is likely that smaller alterations will result in weak X' peaks (due to over smoothing), and thus be missed by the segmentation. To improve sensitivity to small alterations when σ in high, the segmentation and calling algorithms are automatically re-run with $\sigma/2$ and a separate set of fine scale segments is defined.

Finally, the segments are superimposed onto the array elements and a new data column is added for each array element representing the call value for the overlapping segment. Fine scale segments (if present) are merged with the existing segments (defined above) and calls of −2, −1 or +1, 2 are retained only the existing segmentation call for the entire fine scale segment is 0. The array data file is then exported as per user parameters.

Generation and analysis of simulated data

A simulated data set was generated based on the Agilent 244K CGH array mapping, so as to represent the variable element spacing present in real array data. To generate this data, we first mapped random Gaussian noise on to the Agilent 244K CGH array. The standard deviation of the generated distribution was set to equal that of the BT474 breast cancer cell line used below to demonstrate performance on real data (SD = 0.0825), and represents the random noise present in an average array CGH experiment after image extraction and normalization for systematic artefacts. One hundred non-overlapping simulated regions of alteration were then randomly scattered throughout the genome (avoiding centromeric and telomeric regions) for each of the following alterations sizes: 3, 9, 27, 81, 243 and 729 elements (600 total alterations in 6 data sets of 244 000 elements). To simulate the range of alterations expected in a typical CGH experiment we then added a constant $\log_2$ ratio of 0.11, 0.2, 0.4 and 0.8 to each element in each simulated region. These $\log_2$ ratio shifts (deviation from normal) represent alterations ranging from typical low level gains in a heterogeneous sample to high level amplicons. This is also equivalent to an adjustment in SNR. The resulting 20 simulated 244K data files were then analyzed by *DNACopy* with *CGHCall*, *GLAD* and *FACADE* (20–22). This process was repeated three times for a total of 60 simulated 244K data files.

DNACopy with *CGHCall* and *GLAD* were run with default parameters, *FACADE* was run with the following parameters (sigma: 10, baseline distribution: 2 kb, smoother: 5 kb, breakpoints: 10 000, 3 level output: yes, 5 level output: no, remove outliers: yes, $\log_2$ ratio delta: 0.1, *P*-value: = 0.05).

Observed (algorithm results) and expected calls (simulated status) were compared on a per element basis for each alteration size and ratio response (100 alterations of each size per ratio level, representing 300–72 900 altered elements per file). True positives, false positives, true negatives and false negatives were then aggregated for each algorithm and simulation to calculate sensitivity and specificity.

Experimentally derived data

The well-characterized BT474 cell line was previously analyzed with the Agilent 244K CGH array (1). Data can be downloaded from the Gene Expression Omnibus (http://www.ncbi.nlm.nih.gov/geo/) under the accession GSE6415. Parameters for *FACADE*, *CGHCall* and *GLAD* were the same as those used to determine execution times (below).

Execution time

FACADE was run with the following parameters for the 32K BAC array platform: sigma: 6, baseline distribution: 10 kb, smoother: 5 kb—this parameter is not used for tiling BAC platforms, breakpoints: 5000, 3 level output: no, 5 level output: yes, amplification: 0.8, Deletion: −0.5, remove outliers: no, $\log_2$ ratio delta: 0.1, *P*-value: = 0.05

FACADE was run with the following parameters for oligonucleotide platforms (Agilent 244K, Affymetrix 500K and SNP 6.0): sigma: 10, baseline distribution: 2 kb, smoother: 5 kb, breakpoints: 10 000, 3 level output: no, 5 level output: yes, amplification: 0.8, deletion: −0.5, remove outliers: yes, $\log_2$ ratio delta: 0.1, *P*-value: 0.05

All algorithms were run on an Intel Core 2 Quad 2.66 GHz with 4 GB of RAM running Windows XP.

For *DNACopy* with *CGHCall* and *FACADE* execution time was determined using the built in timer (20–22). For *GLAD* we used the R system.time() function. Execution times were determined by three replicate runs for runs under 30 min (execution times >30 min were computed once), and exclude file loading and saving which add a small amount of time to all algorithms.

RESULTS AND DISCUSSION

Unique features of *FACADE* algorithm

FACADE represents an application of a two stage first and second derivative based edge detector in combination with an integrated non-parametric statistical calling and region merging algorithm to identify significant regions of copy number gain and loss. This is significant as many array CGH algorithms tend to offer only segmentation, and called data allows more accurate downstream analysis than segmentation alone, or smoothing based wavelet algorithms (8).

A key challenge in statistical identification of gains and losses is the selection of regions of baseline copy number to test against. In *FACADE*, regions of normal copy number are established by a novel iterative procedure; independent of the segmentation process, where stretches of array elements are selected to have a minimized mean and standard deviation (see 'Materials and Methods' for details). A non-parametric test (Mann–Whitney U-test) is then used to call segments as gained or lost with respect to the samples average ploidy. Additionally our algorithm incorporates the calling of 5 in addition to the option of 3 copy number levels. This enhanced level of calling is significant, as deletions (loss of >1 copy) and amplifications (>5 copies) tend to harbor clinically and biologically relevant genes. Data can be exported as array elements or genes for rapid integration with other software packages and data sets.

Additionally, we have developed a simple graphical user interface that uses standard tab delimited text files as input. These are a common output format of many platform specific data extraction and normalization software packages.

For a complete description of the algorithm underlying *FACADE* please see the 'Materials and Methods' section.

Evaluation of *FACADE* on a high density simulated data set

To demonstrate the accuracy of our algorithm, we compared the relative performance of *FACADE* to that of *GLAD* and *CGHCall* paired with *DNACopy* (two popular array segmentation and calling packages, which have previously been determined to perform with high sensitivity and specificity compared to other popular algorithms) (15,20–23). We have excluded segmentation-only packages from this comparison as we believe called data is critical to interpretation of CGH results. Additionally we have excluded SNP array specific CNV calling algorithms (which are designed specifically to detect small/rare copy number variations in normal diploid samples rather than complex genomes such as those observed in cancer) (16).

To determine the sensitivity and specificity of each algorithm we first generated a simulated Agilent 244K data set to provide a nonbiased comparison of platforms. This data set was generated to simulate 100 alterations each of multiple sizes and ratio shifts within a data set that closely approximates real array CGH data, yet has an established ground truth for regions of alteration. Additionally using a real array mapping density allows analysis in a real world environment where probes are occasionally biased towards denser coverage of some regions.

The results of this analysis are detailed in Figure 1A–F. All algorithms perform well at detecting alterations of any size (spanning 3 elements to 729 elements per segment) with a $\log_2$ ratio shift of at least 0.4. This value is lower than the theoretical ratio for a single copy gain (0.58), owing to the fact that experimental noise and tissue heterogeneity can result in a far lower ratio being observed in many real world samples. Under default parameters, *FACADE* demonstrates the highest sensitivity for small and large size low ratio shift alterations, while *DNACopy* with *CGHCall* demonstrates the highest sensitivity on mid-sized alterations. Both *FACADE* and *CGHCall* with *DNACopy* outperform *GLAD* for all alterations with low ratio shifts. The significant error bars indicated for *CGHCall* in the 27 and 81 element simulations at a $\log_2$ ratio shift or 0.11 are due to a sensitivity of 0 in a single trial for each alteration size. The other two trials demonstrate an average sensitivity of 0.94 (27 element alterations) and 0.98 (81 element alterations), the results were repeatable and derived from separate simulations; thus the reason for the failure to detect any alterations in two independent trials in unclear. Specificity is similar for all algorithms and is acceptably stringent in all cases.

The parameters utilized in Figure 1, are optimized for high sensitivity, as analysis of large sample sets can tolerate spurious false positives. However for cases where higher specificity is required the parameters can be adjusted accordingly (Supplementary Figure S2). Similarly, it is likely that *GLAD* and *CGHCall* may offer improved sensitivity to low SNR events with parameter optimization. However the default parameters demonstrate similar performance among all three algorithms for most alteration sizes and ratio responses. This clearly

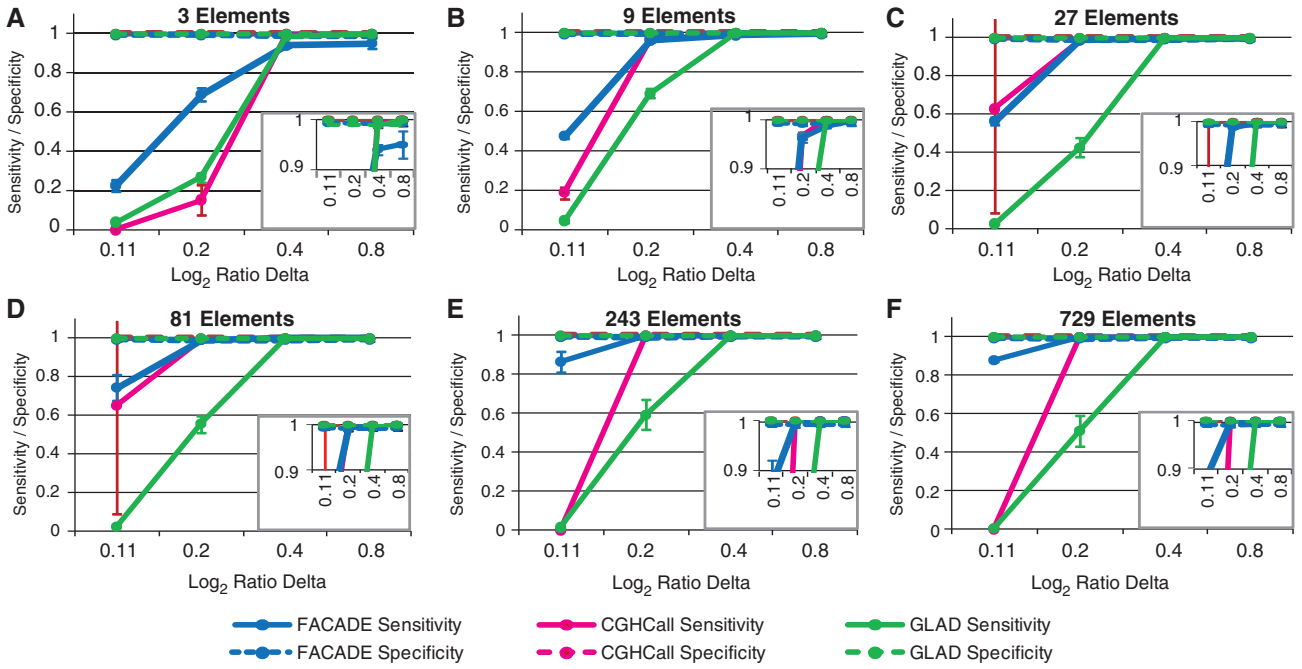


Figure 1. Overview of *FACADE* performance on simulated high resolution data. (**A–F**) Sensitivity and specificity obtained from execution of *FACADE*, and other popular algorithms on a simulated Agilent 244K data set are demonstrated for varying alteration sizes (defined in terms of the number of array elements altered within each of 100 segments per data set) and Log_2 ratio shifts. All algorithms offer similar sensitivity and specificity for alterations at Log_2 ratio shifts of 0.4 or higher. For low level alterations *DNACopy* with *CGHCall* and *FACADE* offer the most robust detection, with *FACADE* offering the best performance on small and large alterations (**A, B, E, F**), and *DNACopy* with *CGHCall* demonstrating higher sensitivity to midsized alterations (**C, D**). Insets highlight the similarity in specificity between all three algorithms.

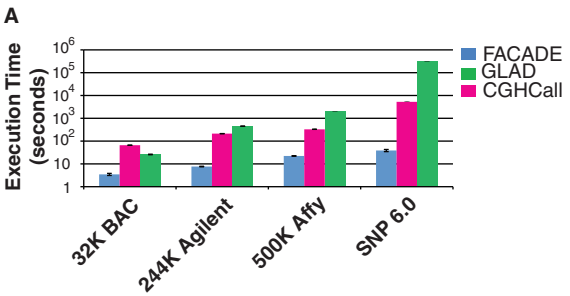
demonstrates the accuracy of *FACADE* as being comparable to established algorithms.

Execution times and algorithm complexity

Array CGH segmentation and calling algorithms tend to be computationally demanding. We have not included segmentation and smoothing only algorithms, which are fast but exclude the calling step. To determine the execution time improvements offered by *FACADE*, we ran *DNACopy* with *CGHCall*, *GLAD* and *FACADE* to segment a complex cell line. For parameters, please see 'Materials and Methods' section.

The algorithm execution times are detailed in Figure 2. Strikingly we achieved a significant (up to several orders of magnitude) decrease in execution times for *FACADE* compared to both *CGHCall* and *GLAD*, in addition to a near linear relationship between execution time and density for high density oligonucleotide platforms. This speed increase is similar to that observed for a recent segmentation only application. However in this study we are processing both segmentation and statistical alteration calling (14).

FAÇADE's rapid execution times will enable users to survey array CGH results from large studies with high accuracy. Scaling our results to a standard sample set of 100 cases using the SNP 6.0 platform would take 1 h to process using *FACADE*, 6 days using *DNACopy* and *CGHCall*, and strikingly 357 days using *GLAD*. This example highlights the real world need for fast accurate algorithms such as *FACADE*.



B

Execution Time in Seconds (+/- 1 sd)

Algorithm	32K BAC	244K Agilent	500K Affy	SNP 6.0
DNA Copy and CGHCall	65.96 +/- 0.06	212.81 +/- 0.1	327.67 +/- 0.02	5266.62
GLAD	25.96 +/- 0.71	450.57 +/-1.06	1999.93	308652.8
FACADE	3.47 +/- 0.37	7.66 +/- 0.2	22.3 +/- 0.59	38.74 +/-4.17
array elements	25916	237009	500448	909508

Figure 2. Execution times of *FACADE* and popular algorithms. Execution times for *FACADE* were compared to the popular algorithms *DNACopy* with *CGHCall* and *GLAD* for several modern array platforms. Execution time is plotted on a log scale, and error bars are indicated for execution times <30 min **(A)**. *FACADE* demonstrates very low execution times compared to both *DNACopy* with *CGHCall* and *GLAD*. This is particularly apparent in high resolution platforms where conventional algorithms exceed reasonable execution times (**A, B**).

Recently Wang *et al.* (11) demonstrated that read depth based sequencing data can be treated similarly to array data and segmented using a similar methodology. Applications such as this will demand fast accurate

algorithms such as *FACADE*. Additionally, *FACADE*'s modular algorithm is conceptually parallelizable and future development can take advantage of this as read depth sequencing increases in coverage allowing smaller bins (for determination of average read depth), and decreases in cost (increased number of samples).

Evaluation of *FACADE* performance on a complex cancer genome

To confirm the results of the simulated data analysis, we expanded our analysis to include the well-characterized BT474 breast cancer cell line (1).

Simulated data does not always reflect the complex alteration patterns of real array data. Thus, we compared the results of our algorithm to *DNACopy* with *CGHCall* and *GLAD* for an Agilent 244K profile of the well characterized breast cancer cell lines BT474 (1). Due to the lack of ground truth we cannot quantify the sensitivity and specificity of each algorithm on the real data. Therefore, to quantify this similarity we compared the algorithms calling performance on clones with $\log_2$ ratios >0.2 (the point in the simulated data where all algorithms' begin to perform acceptably with standard parameters). Consensus calls were generate based on the detection of an alteration by at least two algorithms. For *CGHCall* 98.68% of calls matched the consensus, 0.15% of consensus calls were not detected and 1.16% of call were unique. For *FACADE* 94.12% of calls matched the consensus, 0.54% of consensus calls were not detected, and 5.34% of call were unique. For *GLAD* 64.93% of calls matched the consensus, 34.86% of consensus calls were not detected and 0.21% of call were unique. This fits well with the simulated data results which demonstrated similar performance for *CGHCall* and *FACADE* at log ratios >0.2 (with a sensitivity increase in *FACADE* for small alterations) and reduced sensitivity in *GLAD* (when using default parameters). This is shown in Figure 3A–B, with *FACADE* demonstrating improved sensitivity for low level ratio changes such as that observed on the p-arm of chromosome 11, while retaining a high level of overall accuracy, on par with the performance of established algorithms. These results are highly similar to the simulated data analysis, supporting the application of *FACADE* to complex cancer specimens.

As discussed with the simulated data, *FACADE* can be adjusted to increase either specificity or sensitivity for specific applications depending on the user's needs. For example, we can foresee situations where users are most interested in multi copy gains and homozygous deletions and accordingly, *FACADE* can be detuned for these purposes by simply adjusting *P*-value or ratio thresholds. Similarly, small alterations (which are often copy number polymorphisms) can be filtered out by adjusting the smoother parameters or *P*-value cut offs as needed.

This result strongly demonstrates the high sensitivity of *FACADE*, and ability to detect alterations that are characterized by established algorithms.

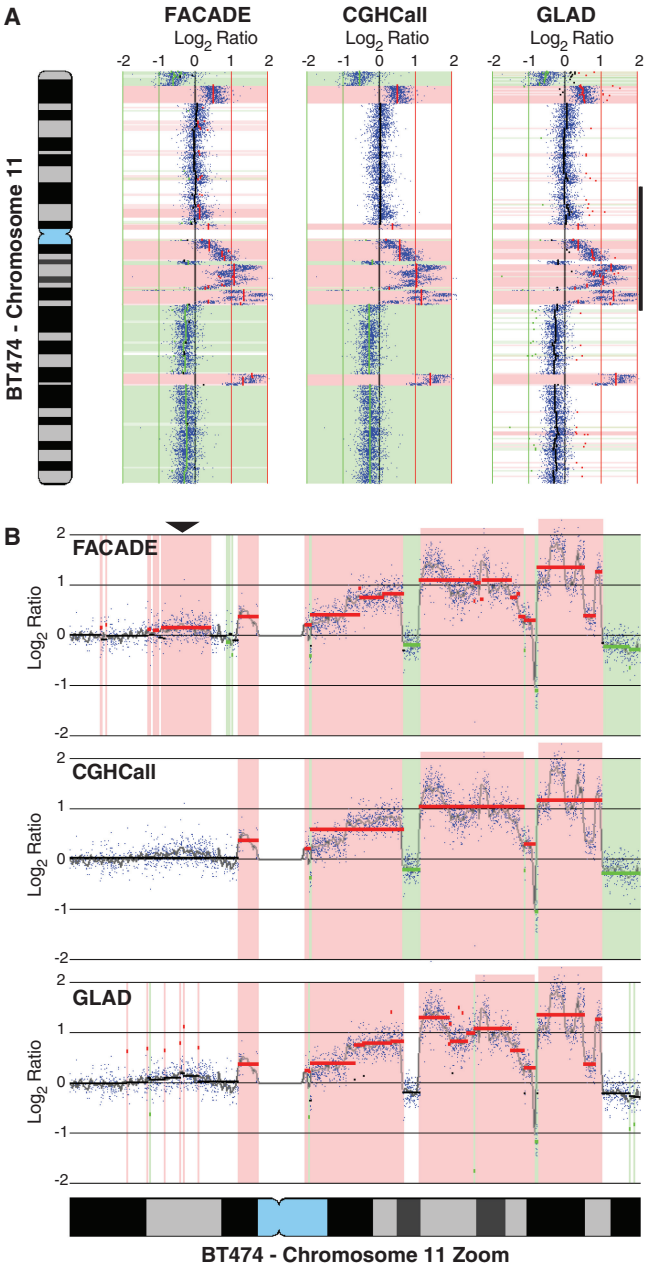


Figure 3. Demonstration of *FACADE* performance on a complex cancer genome. Displayed are the segmentation results from *FACADE*, *DNACopy* with *CGHCall* and *GLAD*, applied to an Agilent 244K profile of the BT474 cell line (displayed as the Log_2 signal ratio of BT474 versus Reference). Shading and colored lines indicate regions detected as copy number gain (red) and loss (green) by each algorithm. The results for chromosome 11 (**A**) clearly demonstrate the similarity in overall segmentation results between all three algorithms, with a slight reduction in deletion detection in *GLAD*, and increase sensitivity to low level gains in *FACADE*. This can be clearly seen in the zoomed view of 11p12 to 11q13 (**B**) where a low level gain (indicated by an arrow) is clearly detected by only *FACADE*.

CONCLUSIONS

We have demonstrated a rapid segmentation and calling algorithm *(FACADE)* that performs competitively with other popular algorithms, while demonstrating rapid execution times which can be orders of magnitude faster than

established algorithms. This is accomplished by utilizing edge detection in combination with non-parametric statistics. Additionally, *FACADE* requires no specialized knowledge from the user, or complex software environments. *FACADE* is designed to handle the next generation high-resolution copy number platforms due to the linear scalability of the algorithm. *FACADE* fills the need, in both research and clinical settings, for rapid accurate segmentation demanded by high-resolution array platforms, large data sets and other situations where long execution times are not tolerable.

FACADE is freely available for academic use, compiled Java (version 1.6 or later) binaries and source code can be obtained from (http://sigma.bccrc.ca/FACADE/). A detailed user manual is provided with the application.

ACKNOWLEDGEMENTS

The authors thank Byron Cline, Craig Wedseltoft, Phillip Wang, Will Lockwood, Chad Malloff and Kim Lonergan for their assistance.

FUNDING

Canadian Institutes of Health Research; Canadian Cancer Society; Canary Foundation; National Institute of Health/ National Cancer Institute Early Detection Research Network; and scholarships from the Canadian Institutes of Health Research; and Michael Smith Foundation for Health Research. Funding for open access charge: Canadian Institutes of Health Research.

Conflict of interest statement. None declared.

REFERENCES

1. Coe,B.P., Ylstra,B., Carvalho,B., Meijer,G.A., Macaulay,C. and Lam,W.L. (2007) Resolving the resolution of array CGH. *Genomics*, **89**, 647–653.
2. Albertson,D.G. and Pinkel,D. (2003) Genomic microarrays in human genetic disease and cancer. *Hum. Mol. Genet.*, **12(Spec No. 2)**, R145–R152.
3. Wong,K.K., deLeeuw,R.J., Dosanjh,N.S., Kimm,L.R., Cheng,Z., Horsman,D.E., MacAulay,C., Ng,R.T., Brown,C.J., Eichler,E.E. *et al*. (2007) A comprehensive analysis of common copy-number variations in the human genome. *Am. J. Hum. Genet.*, **80**, 91–104.
4. Sagoo,G.S., Butterworth,A.S., Sanderson,S., Shaw-Smith,C., Higgins,J.P. and Burton,H. (2009) Array CGH in patients with learning disability (mental retardation) and congenital anomalies: updated systematic review and meta-analysis of 19 studies and 13,926 subjects. *Genet. Med.*, **11**, 139–146.
5. Redon,R., Ishikawa,S., Fitch,K.R., Feuk,L., Perry,G.H., Andrews,T.D., Fiegler,H., Shapero,M.H., Carson,A.R., Chen,W. *et al*. (2006) Global variation in copy number in the human genome. *Nature*, **444**, 444–454.
6. Edelmann,L. and Hirschhorn,K. (2009) Clinical utility of array CGH for the detection of chromosomal imbalances associated with mental retardation and multiple congenital anomalies. *Ann. NY Acad. Sci.*, **1151**, 157–166.
7. Garnis,C., Coe,B.P., Lam,S.L., MacAulay,C. and Lam,W.L. (2005) High-resolution array CGH increases heterogeneity tolerance in the analysis of clinical samples. *Genomics*, **85**, 790–793.
8. van Wieringen,W.N., van de Wiel,M.A. and Ylstra,B. (2007) Normalized, Segmented or Called aCGH? *Cancer Informatics*, **3**, 321–327.
9. Snijders,A.M., Schmidt,B.L., Fridlyand,J., Dekker,N., Pinkel,D., Jordan,R.C. and Albertson,D.G. (2005) Rare amplicons implicate frequent deregulation of cell fate specification pathways in oral squamous cell carcinoma. *Oncogene*, **24**, 4232–4242.
10. Bredel,M., Bredel,C., Juric,D., Harsh,G.R., Vogel,H., Recht,L.D. and Sikic,B.I. (2005) High-resolution genome-wide mapping of genetic alterations in human glial brain tumors. *Cancer Res.*, **65**, 4088–4096.
11. Wang,L.Y., Abyzov,A., Korbel,J.O., Snyder,M. and Gerstein,M. (2009) MSB: a mean-shift-based approach for the analysis of structural variation in the genome. *Genome Res.*, **19**, 106–117.
12. Stjernqvist,S., Ryden,T., Skold,M. and Staaf,J. (2007) Continuous-index hidden Markov modelling of array CGH copy number data. *Bioinformatics*, **23**, 1006–1014.
13. Huang,H., Nguyen,N., Oraintara,S. and Vo,A. (2008) Array CGH data modeling and smoothing in Stationary Wavelet Packet Transform domain. *BMC Genomics*, **9(Suppl. 2)**, S17.
14. Ben-Yaacov,E. and Eldar,Y.C. (2008) A fast and flexible method for the segmentation of aCGH data. *Bioinformatics*, **24**, i139–i145.
15. Pique-Regi,R., Monso-Varona,J., Ortega,A., Seeger,R.C., Triche,T.J. and Asgharzadeh,S. (2008) Sparse representation and Baysian detection of genome copy number alterations from microarray data. *Bioinformatics*, **24**, 309–318.
16. Winchester,L., Yau,C. and Ragoussis,J. (2009) Comparing CNV detection methods for SNP arrays. *Brief Funct. Genomic Proteomic*, **8**, 353–366.
17. Diaz-Uriarte,R. and Rueda,O.M. (2007) ADaCGH: a parallelized web-based application and R package for the analysis of aCGH data. *PLoS ONE*, **2**, e737.
18. Canny,J. (1986) A computational approach to edge detection. *IEEE Trans. Patt. Anal.Machine Intelligence*, **PAMI-8**, 679–698.
19. Marr,D. and Hildreth,E. (1980) Theory of edge detection. *Proc. Roy. Soc. Lond.*, **207**, 187–217.
20. Venkatraman,E.S. and Olshen,A.B. (2007) A faster circular binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics*, **23**, 657–663.
21. van de Wiel,M.A., Kim,K.I., Vosse,S.J., van Wieringen,W.N., Wilting,S.M. and Ylstra,B. (2007) CGHcall: calling aberrations for array CGH tumor profiles. *Bioinformatics*, **23**, 892–894.
22. Hupe,P., Stransky,N., Thiery,J.P., Radvanyi,F. and Barillot,E. (2004) Analysis of array CGH data: from signal ratio to gain and loss of DNA regions. *Bioinformatics*, **20**, 3413–3422.
23. Lai,W.R., Johnson,M.D., Kucherlapati,R. and Park,P.J. (2005) Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data. *Bioinformatics*, **21**, 3763–3770.