

Software

Open Access

Caryoscope: An Open Source Java application for viewing microarray data in a genomic context

Ihab AB Awad¹, Christian A Rees¹, Tina Hernandez-Boussard², Catherine A Ball² and Gavin Sherlock*¹

Address: ¹Dept. of Genetics, 300 Pasteur Drive, Stanford University Medical School, Stanford, CA 94305-5120, USA and ²Dept. of Biochemistry, Stanford University School of Medicine, Stanford, CA 94305-5307, USA

Email: Ihab AB Awad - ihab@stanford.edu; Christian A Rees - rees@genome.stanford.edu; Tina Hernandez-Boussard - boussard@genome.stanford.edu; Catherine A Ball - ball@genome.stanford.edu; Gavin Sherlock* - sherlock@genome.stanford.edu

* Corresponding author

Published: 15 October 2004

Received: 12 May 2004

BMC Bioinformatics 2004, 5:151 doi:10.1186/1471-2105-5-151

Accepted: 15 October 2004

This article is available from: <http://www.biomedcentral.com/1471-2105/5/151>

© 2004 Awad et al; licensee BioMed Central Ltd.

This is an open-access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/2.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Background: Microarray-based comparative genome hybridization experiments generate data that can be mapped onto the genome. These data are interpreted more easily when represented graphically in a genomic context.

Results: We have developed Caryoscope, which is an open source Java application for visualizing microarray data from array comparative genome hybridization experiments in a genomic context. Caryoscope can read General Feature Format files (GFF files), as well as comma- and tab-delimited files, that define the genomic positions of the microarray reporters for which data are obtained. The microarray data can be browsed using an interactive, zoomable interface, which helps users identify regions of chromosomal deletion or amplification. The graphical representation of the data can be exported in a number of graphic formats, including publication-quality formats such as PostScript.

Conclusion: Caryoscope is a useful tool that can aid in the visualization, exploration and interpretation of microarray data in a genomic context.

Background

The application of high-throughput technologies (such as DNA microarrays) to biomedical experimentation generates large quantities of data that can be difficult to browse and interpret in the absence of a graphical representation. Eisen *et al.* have previously displayed clustered microarray data using a false color representation that greatly aids in the intuitive interpretation of the data ([1]). However, when these data are from array comparative genome hybridization (arrayCGH) experiments (*e.g.*, see [2]), the genomic locations of the reporters (the molecules in each

spot on a microarray) that were used to generate the data are important for interpretation. A relative increase or decrease in the ratios for a group of reporters that report on adjacent genomic locations may indicate amplification or deletion of that genomic region, respectively. Additionally, even the analysis of expression data in the context of genomic position can also identify regions of amplification or deletion, or even cases of aneuploidy ([3,4]).

In addition to being able to view and browse arrayCGH data, it is also important that the data be readily

connected to annotation sources, such that a user can easily determine the identity and attributes of the gene represented by a reporter that was present on a microarray, which for instance may show evidence of amplification or deletion. For example, in arrayCGH experiments using tumor cells as the DNA source, there is an obvious value in rapidly determining whether a deleted region contains a tumor suppressor gene.

Finally, researchers frequently need to create figures, for publications, communication with co-workers, supplemental websites, or presentations. Thus researchers should be able to produce the visual representation of their data in a variety of graphic formats.

Caryoscope was originally implemented as a Web form, generating either a bitmap or a clickable PDF output. When this became an important day-to-day tool for our users ([5-9]), we created an improved, interactive version, consisting of a standalone application for analyzing arrayCGH data and an open architecture of re-usable classes that may be embedded by other developers in their own applications. In this paper, we focus on Caryoscope as an application.

Some other software packages were developed while this work was in progress, and can perform some of these functions. For instance, Genome2D ([10]) is designed to display bacterial transcriptome data on linear chromosome maps, while SeeCGH ([11]) was designed for viewing arrayCGH data (only for 2-channel arrays). However, both of these programs are designed to run solely on the Windows operating system, whereas Caryoscope is a Java application that can be used on Macintosh OS X, Linux and various UNIX operating systems, as well as Windows. Greshock *et al.* ([12]) have built similar functionality, called CGHAnalyzer, on TIGR's Multiple Experiment Viewer (MeV) platform, but with a different (circular) whole-genome view. Furthermore, Caryoscope can be run in a command line mode, making it easy to embed within a CGI or a processing pipeline.

Implementation

We implemented Caryoscope in Java ([13]) and deployed it as a Java Web Start ([14]) application, so a user may run it directly from our website ([15]) by clicking on a link. One can also install Caryoscope directly on a computer, but we recommend launching via the website in order to obtain the most current version of the software.

Caryoscope accepts data as text input files in simple formats so as to maximize interoperability with other systems.

Results

Application features

As input, Caryoscope accepts a single file in either the General Feature Format (GFF, [16]) or a tab-delimited (TXT) or comma-delimited (CSV) spreadsheet-compatible format. This file describes the chromosomes to be displayed, and a set of loci on the chromosomes annotated by a number of associated microarray datasets and other descriptive information. The structure of a Caryoscope input file is illustrated in Figure 1.

Once the user opens a file, Caryoscope automatically displays one of the datasets contained therein (Figure 2). Caryoscope displays each feature as a rectangle on the chromosome axis; the size of the rectangle on the horizontal axis, perpendicular to the chromosome, represents the magnitude of the associated data value, while the size of the rectangle in the vertical direction, along the chromosome axis, represents the size of the represented feature, based on its genomic coordinates. Pursuant to convention, the default display of Caryoscope represents positive values in red bars, which are drawn to the right of the chromosome, and negative values in green bars (though these colors can be changed), which are drawn to the left of the chromosome. Thus, based on color, size, and location, researchers can easily intuit the meaning of the graphical representation of their data.

Caryoscope provides several modes in which the user may view the data; these are controlled by the **View modes** toolbar (Figure 2). In the various panning and zooming modes, the user may change the view of the data to drill down to specific regions of interest. In **Navigate** mode, the user sees tooltips (small informational pop-up windows) that appear immediately when mousing over the features, and can navigate to related URLs by clicking on each feature. Typically, users at Stanford University (our primary source of testers and users) link GenBank accessions, associated with the cDNA clones that are on their microarrays, to SOURCE Gene Pages ([17]).

The zooming paradigm in Caryoscope is somewhat novel in that it permits independent control of the zoom scales in the X and Y directions (Figure 3). It allows users to select the best scaling to see detail along the chromosome axis, and the data values perpendicular to the chromosome axis, for their specific data. The **Reset viewpoint** button on the **View modes** toolbar (Figure 2) allows the user to return quickly to the default scaling.

The behavior of Caryoscope in **Navigate** mode is shown in Figure 4. The tooltip and URL text are computed for each feature by substituting the value of its annotations into the **Feature tooltip expression** and **Feature URL**

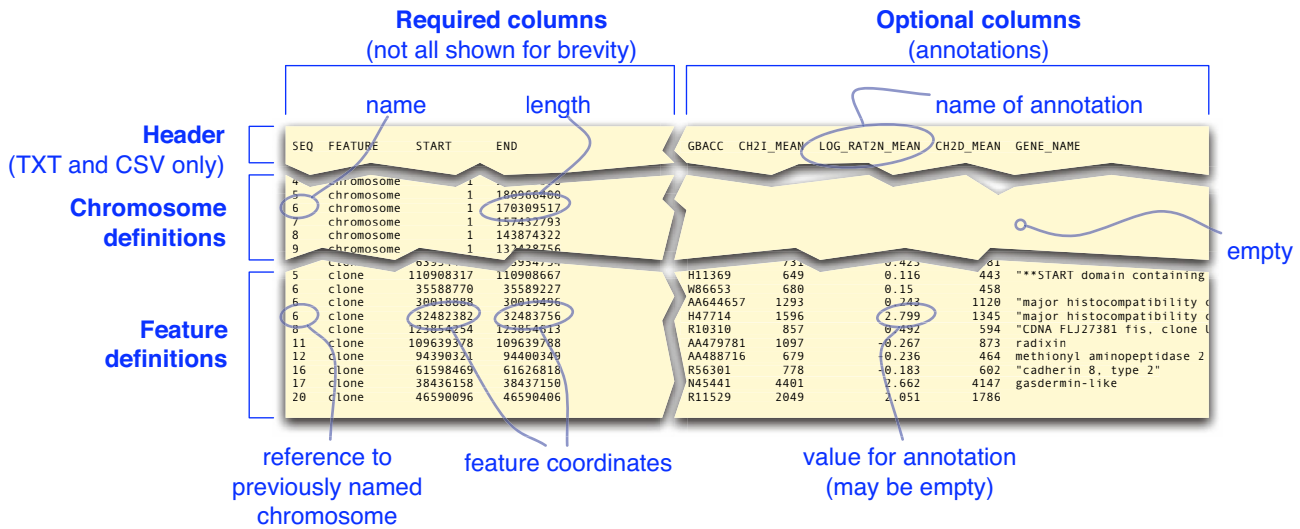


Figure 1
The Caryoscope input file format. An illustration of the input file format for Caryoscope. A tab-delimited (TXT) file is shown as an example, but the CSV and GFF formats are similar. Each column of data represents an annotation, with the name of the annotation at the top. Note how, at this level, there is no distinction between expression data and other information: for maximum flexibility, everything is an annotation.

expression settings, as illustrated in the figure. These features allow users to have immediate access to information about each feature as they browse the data.

The user can enable two built-in computations on the data values: a user can compute the logarithm of the values (to any base specified by the user), and a user can compute a moving average of the values. Both these computations can be controlled from the **Settings** dialog (see Figure 2b). Users can perform other computations outside Caryoscope; this is facilitated by the fact that we support common spreadsheet-compatible file formats (TXT and CSV).

To prepare diagrams, the user can export the Caryoscope display to a variety of graphics formats via the **Export** dialog. Specifically, Caryoscope supports vector (e.g., PostScript and PDF) output for scalable publication-quality results, and raster (e.g., JPEG and PNG) output for ease of viewing, posting on supplemental websites, and inclusion in presentations.

A user may export graphics from Caryoscope via the command line mode, without having to invoke the interactive user interface. For example, to export a view of a dataset as a PDF file, the user could invoke Caryoscope as follows:

```
java -jar caryoscope-run.jar
    -inputFile 3395-2004-04-04.csv
    -visibleDataset RAT2_MEAN
    -outputFile 3395.pdf
    -outputFormat pdf
```

All settings may be modified via the command line. The `-listFormats` option provides a list of available graphics output formats, and the `-help` option prints a brief summary of the options.

A comparison of the features of Caryoscope with other, similar software is presented in Table 1.

Obtaining Caryoscope

In addition to immediately executable copies, the complete source code for Caryoscope is available without limitations from our website ([15]), and is covered by a very liberal Open Source ([18]) license (the MIT License, [19]). All external components used by Caryoscope are also Open Source.

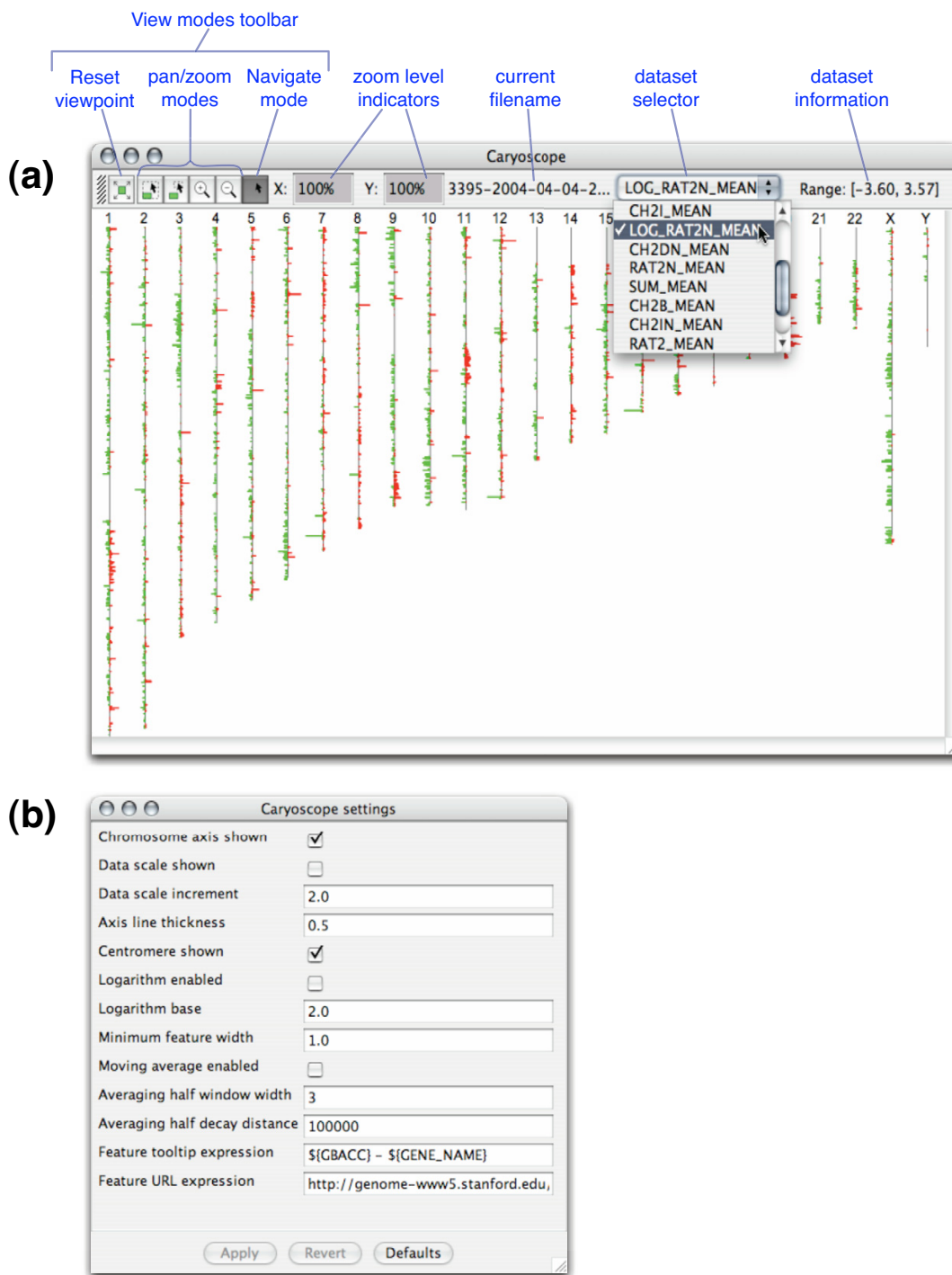


Figure 2

The Caryoscope main window. (a) Caryoscope displays all numerical annotations from the data file as datasets. The datasets are available for viewing via the dataset selector at the top of the Caryoscope window. (The data shown are provided by Jon Pollack, from a breast tumor sample. Note the significant regions of amplification and deletion.) The toolbar at the top allows the user to select a variety of modes for viewing the data. The panning and zooming modes allow the user to navigate the data, while the **Reset viewpoint** button returns to display all of the data in the current dataset, scaled to the current window size. The **Navigate** mode enables hyperlinking and tooltips. The zoom level indicators improve the user's situational awareness while viewing the data. (b) The **Settings** window allows the user to configure the application.

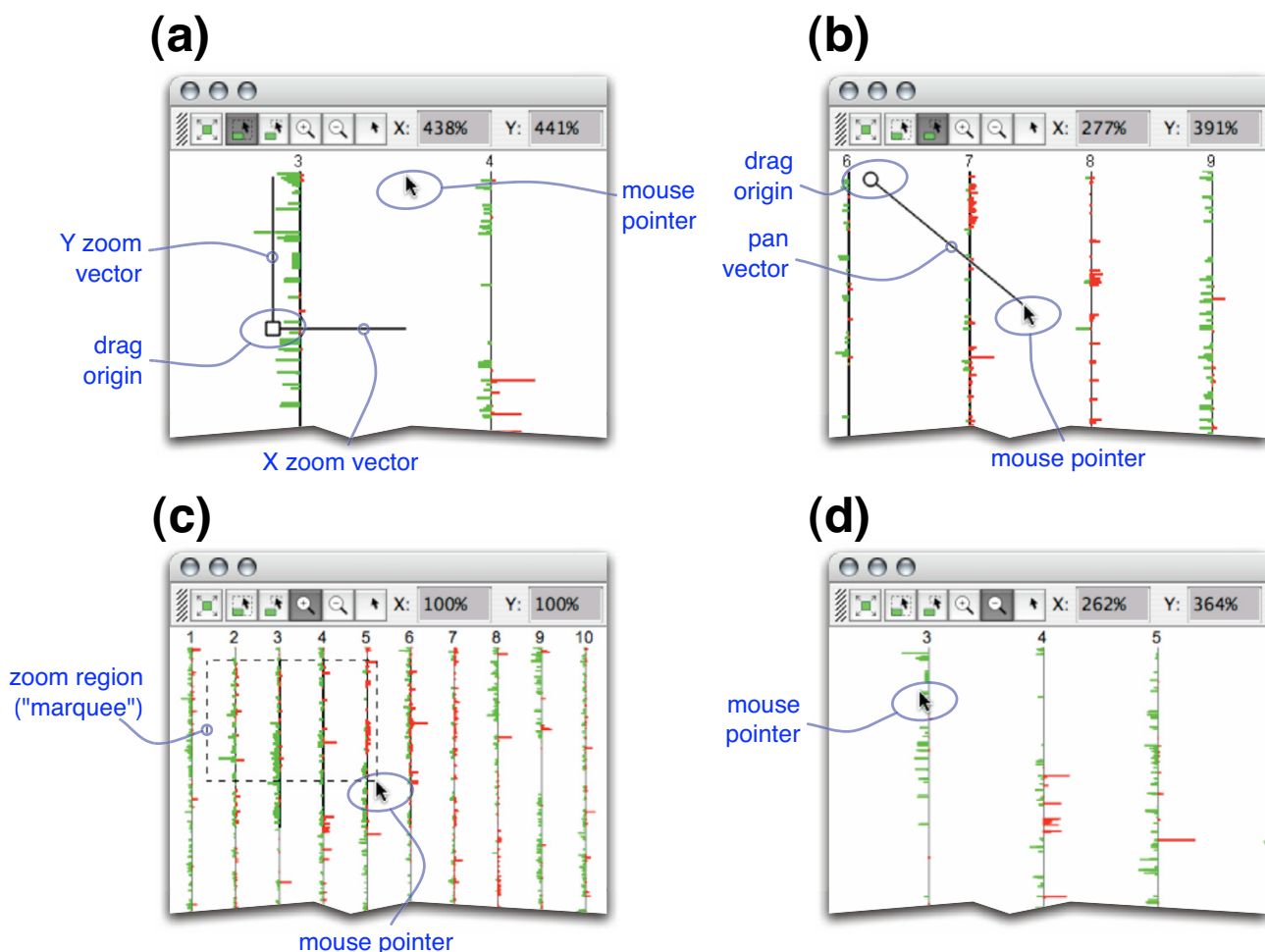


Figure 3
Zooming and panning modes. The zooming and panning modes are selected from the **View modes** toolbar. (a) In **Dynamic zoom** mode, the user clicks then drags the mouse; based on the mouse motion, the view zooms continuously with independent zoom speeds in the X and Y directions. (b) In **Dynamic pan** mode, the user clicks then drags the mouse; the direction and distance of the mouse motion determines the speed and direction of the panning motion through the data. (c) In **Zoom in** mode, the user selects a region (denoted by a "marquee"). The display is zoomed so that the selected region is fit to the current window. The user may also click anywhere within the current window, which zooms in by a fixed scaling, centered on the click point. (d) In **Zoom out** mode, the user clicks anywhere, which zooms out by a fixed scaling, centered on the click point.

We update Caryoscope frequently (approximately once every three weeks) and post news items on the website. We also send e-mail announcements to people who have requested them.

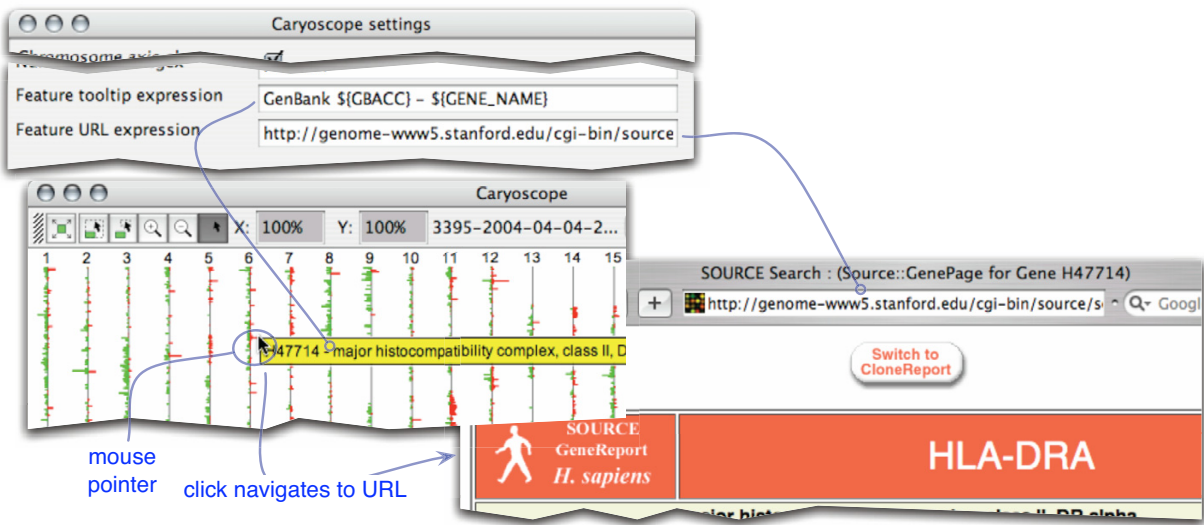
Discussion
Biological context

Caryoscope is useful for viewing both arrayCGH and expression data in the context of genomic position. It helps a biologist gain insight by providing a high-level

view of a large amount of data at once, where patterns can be perceived at a glance.

A biologist studying amplifications or deletions in tumor cells may create and export graphics representing array-CGH and expression data for the same cells using Caryoscope, and visually compare the two side-by-side. For instance, co-located regions that are amplified at the DNA level and over-expressed at the RNA level would provide

(a)



(b)

	Expression	Result
Feature tooltip	GenBank $\${GBACC}$ - $\${GENE_NAME}$	GenBank H47714 - major histocompatibility co...
Feature URL	http://genome-...&criteria= $\${GBACC}$ &choice=Gene	http://genome-...&criteria= H47714 &choice=Gene

SEQ	FEATURE	START	END	GBACC	CH21_MEAN	LOG_RAT2N_MEAN	CH20_MEAN	GENE_NAME
6	clone	35588770	35589227	H11369	732	-0.423	401	***START domain containing
6	clone	30018888	30019496	W80653	680	0.15	458	
6	clone	32482382	32483756	AA64657	1293	0.743	1170	"major histocompatibility c
6	clone	32482382	32483756	H47714	1596	2.799	1345	"major histocompatibility c
8	clone	123834254	123834813	K10510	857	0.492	394	"CDW4 P127361" rns. clone
11	clone	109639378	109639788	AA479781	1097	-0.267	873	radixin
12	clone	94390321	94400349	AA488716	679	-0.236	464	methionyl aminopeptidase 2
16	clone	61598469	61626818	R56301	778	-0.183	602	"cadherin 8, type 2"
17	clone	38436158	38437150	N45441	4401	2.662	4147	gasdermin-like
20	clone	46590096	46590406	R11529	2049	2.051	1786	

Figure 4

Navigate mode. (a) In this mode, Caryoscope shows a tooltip (small informational pop-up window) whenever the user moves the mouse over a feature. When the user clicks on the feature, Caryoscope navigates to a URL related to the feature. Both the tooltip text and the URL are generated dynamically from the annotations on the feature under the mouse pointer, via the **Feature tooltip expression** and **Feature URL expression** settings. (b) Illustration of how the tooltips and URLs are computed from the **Feature tooltip expression** and **Feature URL expression** settings and the annotations in the input file.

excellent confirmation of the results. Using the zooming and panning features of Caryoscope, the biologist could focus on specific regions of interest.

To identify regions of aneuploidy, the biologist can again simply examine the data visually. In this case, however, one would look for a large-scale pattern. One might specify a **Minimum feature width** of, say, 2 pixels (Figure 5c), to ensure that any deletion or amplification, no matter

how small in genomic coordinates, is easily visible. Rather than zooming in on specific regions, a researcher would tend to compare overall views of the entire genome. If it seems like practically all of one chromosome is amplified or deleted, the biologist would have strong evidence for aneuploidy.

In a gene expression study, a biologist may suspect that some expression patterns are correlated with genomic

Table 1: Comparison to other tools Itemized comparison of the features of Caryoscope ([15]), CGHAnalyzer ([12],[22]) and SeeCGH ([11],[23]).

	Caryoscope	CGHAnalyzer	SeeGH
Platform	Java (runs on any popular platform)	Java (runs on any popular platform)	Microsoft Windows
Application availability	Unlimited use; no signup	Unlimited use with signup	Free with signup for academics
Installation process	Install Java; run application directly from Web	Install Java; install application	Install MySQL; install application; configure database tables
Input file formats	Single input file; spreadsheet compatible text or General Feature Format (GFF)	Multi-component input; text-based input formats	Text-based input formats
Internal data format	Text file	Text file or database	MySQL database
Source code availability	Free download	Download with signup	Not available
Source code license	Open Source	Unknown	N/A
Internal code dependencies	All Open Source	Depends on some commercial software	N/A
Superimposed karyotype plot	No	Yes	Yes
Informational summary for each feature	Yes, configurable in application settings	Yes, fixed tabular format	Yes, fixed tabular format
Web link from each feature	Yes, configurable in application settings	Yes, fixed Ensembl or GoldenPath links	No
Data points in genome coordinates	Yes	Yes	Yes
Data points in linear order	No	Yes	No
Simultaneous display of multiple experiments	No	Yes	No
Zoomable display	Yes	No	Yes
Part of a more complete analysis framework	No	Yes (TIGR MeV)	No
Built-in data point filtering	No	No	Yes
Whole-genome view	Linear chromosomes	Concentric circles representing chromosomes	Linear chromosomes
Generate graphics from command line	Yes	No	No

position. Caryoscope allows one to view expression data, either for the whole genome or on a region of interest, to help confirm or refute a hypothesis.

Finally, in all this work, the biologist may want to have quick and easy access to information about the genomic features displayed. As long as the information needed is available from the annotations that were saved in the input data file (or available at a URL that can be built based on the annotations), one can use the **Feature URL expression** and **Feature tooltip expression** (Figure 4) settings to provide immediate mouse-over feedback with this information – almost as if the application were customized for a specific field of interest.

Software context

We intend Caryoscope to be a bench-top visualization tool that biologists can use immediately to get day-to-day work done, with a very low "cost of entry" for getting started. This led us to a number of design choices.

Caryoscope can be launched from our web site without a prior installation step. Since it is Open Source, anyone, including any for-profit organization, can use it without restrictions and without having to obtain a license or register for access.

The input file formats and output graphical formats we chose are all in common usage. In particular, the TXT and CSV input formats can be generated using any popular spreadsheet or database software, or even with a plain text editor, without having to do any programming.

Finally, we built Caryoscope to be content-neutral, with no hard-coded specificity to any research field. Thus, users of Caryoscope may control how annotations are treated as numerical data, and can "program" data-driven interactive behavior of the display (*i.e.*, the tooltips and hyperlinks).

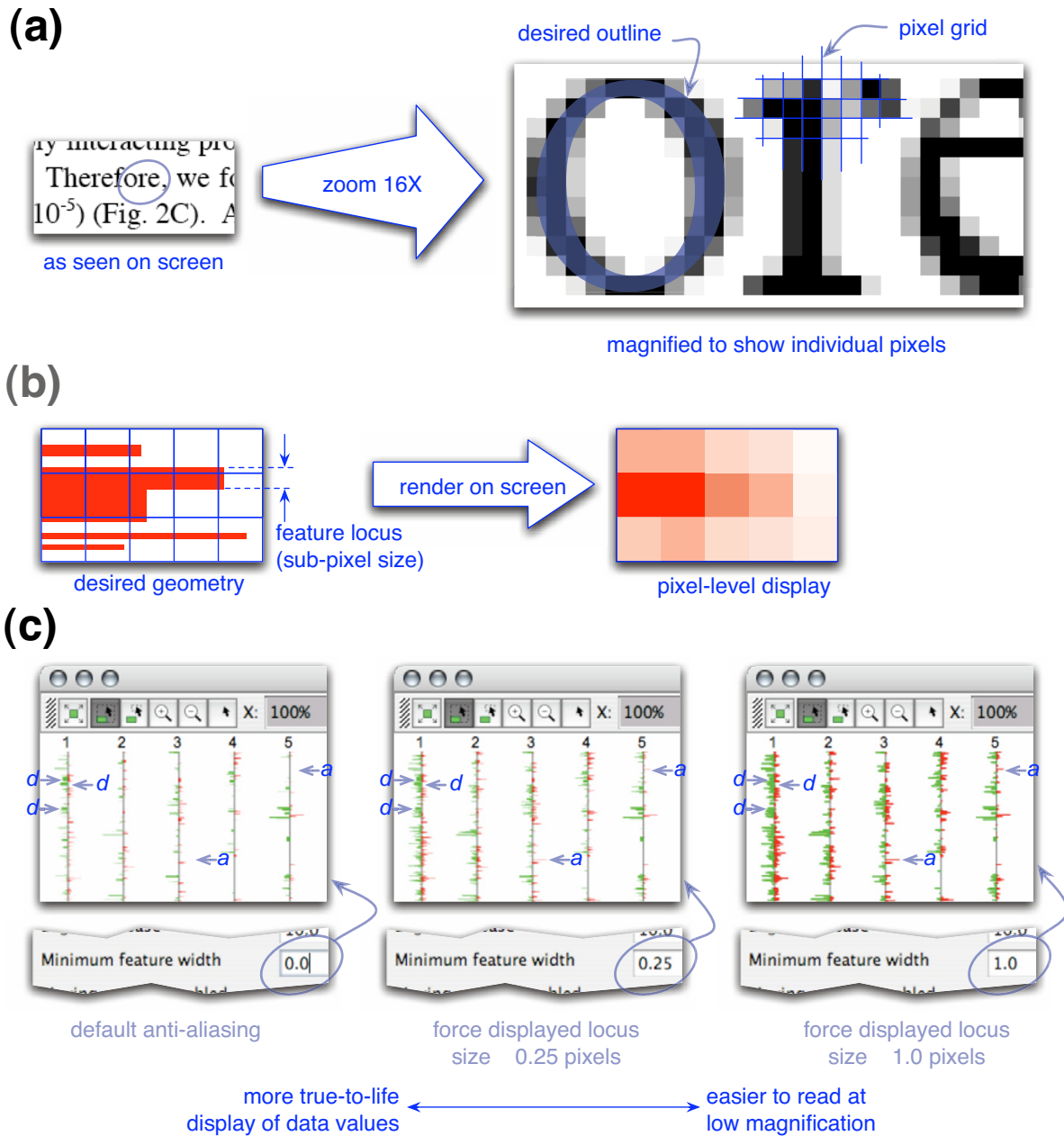


Figure 5

Example of anti-aliasing. A demonstration of the effect of anti-aliasing in Caryoscope. (a) Anti-aliased text displayed by Adobe Acrobat 6.0 on an Apple computer running Macintosh OS 10.3.3. Note that the text appears as if drawn by smooth black ink but, upon magnification, it becomes clear that this smoothness is a visual illusion created by careful manipulation of the color of each pixel. (b) The analogous situation in Caryoscope, where the scaled sizes of the feature loci are smaller than one pixel on the display device. In this example, we applied a form of anti-aliasing by *prefiltering*, where we computed the color of each pixel based on the area of red rectangles it contains. See [20]. (c) Actual displays from Caryoscope showing how the **Minimum feature width** setting highlights different aspects of the data. The three deletions **d** on chromosome 1 are clearly distinct at a setting of zero, but are lost in the midst of other artificially expanded features when the setting is increased. On the other hand, the two amplification "spikes" **a** on chromosomes 3 and 5 are invisible at a setting of zero, but become visible when the setting is increased.

Future work

From the outset of this project, our biggest challenge has been how to accurately represent the huge amounts of data in a typical gene expression or arrayCGH experiment using the limited number of pixels available on the screen or on a printed page. If we skew our display algorithms too much towards producing a "sharp", high-contrast plot, we risk obscuring detail in the data and leading biologists to the wrong conclusions. On the other hand, since the size of the data elements, properly scaled from genomic coordinates to the display device, can be far less than the size of one pixel, we need a supportable way to "summarize" the data within each pixel and represent that summary as a single value: the color (including the brightness) of that pixel.

Modern computer graphics systems use a technique called "anti-aliasing" ([20] and Figure 5a) to render sub-pixel details with the illusion of smoothness. The Java subsystems we use in Caryoscope do this automatically and, in the current version of the software, we simply rely on them (Figure 5b). However, the anti-aliasing in Java is designed to display visually appealing text, lines and arcs, but not to ensure the most accurate possible on-screen rendering of scientific data. Specifically, at low magnification, the data almost disappear unless we force a minimum pixel size for each locus (Figure 5c).

We will develop our display methods further to ensure that we can provide an easy-to-read display while retaining the subtle variations in the data. Following the spirit of medical diagnostic imaging (DICOM, [21]), whereby incorrect details in a few pixels could lead to an incorrect conclusion, we must ensure that our displays, which are used for important research decisions, are never misleading.

One solution, suggested by [12], is to display the data elements, not aligned to the position of the loci along the chromosome, but rather in strict sequential order with a fixed width. While this solves the anti-aliasing problems, it does eliminate consistent chromosomal positions and alignments of the data. Furthermore, it causes the appearance of the display to be dependent on the specific choice of clones – which can be another source of subtle variation when comparing multiple datasets. Another idea is to display dots, rather than horizontal bars, so that the "spread" of the data is more visible even if data points are super-imposed. We will investigate this for a future release of Caryoscope.

We are particularly concerned about the use of Caryoscope (or similar) graphics in vector formats (such as PostScript, PDF and SVG) that are subsequently rendered on diverse display devices and printers. Since we have no

control over the rendering at the destination, it is likely that the same vector output could look very different on different devices. Once we have studied this problem in more detail, we intend to provide practical usage guidelines for researchers.

Our experience with the application, and how it is used, leads us to believe that the current zooming paradigm should be revisited. While the model of a continuously zoomable 2D space provides users with the features they need, it can lead to displays whereby the data in specific regions of adjacent chromosomes are juxtaposed, even though their being next to each other is not intrinsically meaningful (see Figure 6). (An exception to this might occur in telomere amplification, or perhaps special behavior around the centromeres. In this case, the ability to align the chromosomes at either end, or at the centromeres, would be helpful.)

In the future, we will modify our display so that the user can turn "on" or "off" the display of the available chromosomes. Within that display, and with the help of our users, we will review the role of the X-axis scaling: perhaps it should change the scaling of the data, or perhaps it does not belong in Caryoscope-like applications at all.

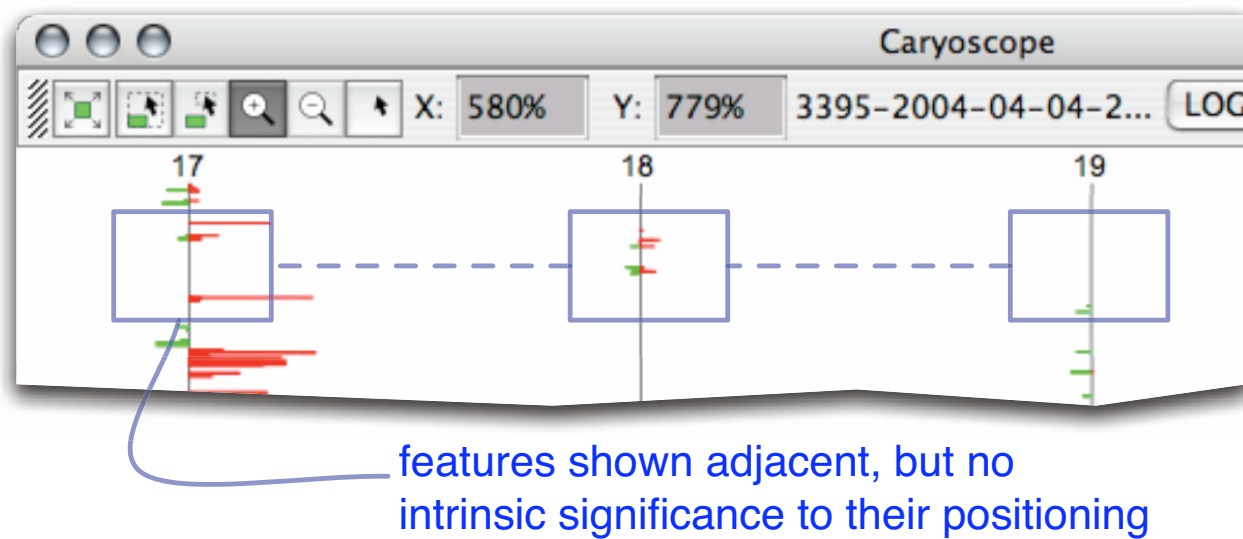
A common user request is to display two or more array-CGH and/or expression datasets side by side, either to determine regions of recurrent deletion or amplification, or to discern visually the impact that changes in chromosome copy number may have on transcript levels (*e.g.*, see [5]). Another frequent request is to show the cytoband information for each chromosome. We intend to add these features as part of our future development, in the course of which we would perhaps redefine, or extend, the manner in which our input data are defined (*i.e.*, we would accept the definition of the chromosome names, lengths and cytobands in a separate file that would be reused by different datasets).

Summary

Caryoscope currently provides a flexible method to visualize, explore and create images of microarray data in a genomic context. With such a tool, microarray researchers will be able to answer questions about how genome copy number or genome position plays a role in biological processes or human diseases.

Conclusions

Caryoscope is a useful, flexible Java application for the visualization of microarray data in a genomic context. It is available as Open Source under the permissive MIT License, allowing anyone to use or modify it.

**Figure 6**

Zooming paradigm. An example of a partially zoomed-in display, which highlights adjacent regions of four chromosomes. The model of continuous 2-dimensional zooming breaks down in this case, since the adjacency of these regions is not biologically relevant.

Availability and requirements

Project name: Caryoscope

Project home page: <http://caryoscope.stanford.edu/>

Operating system(s): Platform independent

Programming language: Java

Other requirements: Java 1.4 or higher

License: MIT License

Any restrictions to use by non-academics: None

Authors' contributions

IABA compiled the functional requirements for the Java version of Caryoscope and designed and implemented the software. CAR conceived and wrote the original Perl version of Caryoscope. THB aided and participated in the design of the study, dealt with user support and worked with users to get feedback and suggestions for further development. CAB aided and participated in the design and evaluation of the study and provided feedback and suggestions for future development. GS supervised and participated in the design of the study. All authors read and approved the final version of the manuscript.

Acknowledgements

The authors would like to thank members of the Pollack, Brown and Botstein laboratories, particularly Jon Pollack, for their feedback on Caryoscope, and for providing datasets for testing, and Ash Alizadeh for suggesting the Caryoscope name. Thanks also go to all the members of the Stanford Microarray Database group for stimulating discussions. This work was funded by a grant from the NHGRI, R01HG002732, to GS.

References

1. Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns.** *Proc Natl Acad Sci U S A* 1998, **95**:14863-14868.
2. Pollack JR, Perou CM, Alizadeh AA, Eisen MB, Pergamenschikov A, Williams CF, Jeffrey SS, Botstein D, Brown PO: **Genome-wide analysis of DNA copy-number changes using cDNA microarrays.** *Nat Genet* 1999, **23**:41-46.
3. Crawley JJ, Furge KA: **Identification of frequent cytogenetic aberrations in hepatocellular carcinoma using gene-expression microarray data.** *Genome Biol* 2002, **3**:RESEARCH0075.
4. Hughes TR, Roberts CJ, Dai H, Jones AR, Meyer MR, Slade D, Burchard J, Dow S, Ward TR, Kidd MJ, Friend SH, Marton MJ: **Widespread aneuploidy revealed by DNA microarray expression profiling.** *Nat Genet* 2000, **25**:333-337.
5. Pollack JR, Sorlie T, Perou CM, Rees CA, Jeffrey SS, Lonning PE, Tibshirani R, Botstein D, Borresen-Dale AL, Brown PO: **Microarray analysis reveals a major direct role of DNA copy number alteration in the transcriptional program of human breast tumors.** *Proc Natl Acad Sci U S A* 2002, **99**:12963-12968.
6. Baldus CD, Liyanarachchi S, Mrozek K, Auer H, Tanner SM, Guimond M, Ruppert AS, Mohamed N, Davuluri RV, Caligiuri MA, Bloomfield CD, de la Chapelle A: **Acute myeloid leukemia with complex karyotypes and abnormal chromosome 21: Amplification discloses overexpression of APP, ETS2, and ERG genes.** *Proc Natl Acad Sci U S A* 2004, **101**:3915-3920.
7. Linn SC, West RB, Pollack JR, Zhu S, Hernandez-Boussard T, Nielsen TO, Rubin BP, Patel R, Goldblum JR, Siegmund D, Botstein D, Brown PO, Gilks CB, van de Rijn M: **Gene expression patterns and gene**

- copy number changes in dermatofibrosarcoma protuberans.** *Am J Pathol* 2003, **163**:2383-2395.
8. Dunham MJ, Badrane H, Ferea T, Adams J, Brown PO, Rosenzweig F, Botstein D: **Characteristic genome rearrangements in experimental evolution of *Saccharomyces cerevisiae*.** *Proc Natl Acad Sci U S A* 2002, **99**:16144-16149.
 9. Lin JY, Pollack JR, Chou FL, Rees CA, Christian AT, Bedford JS, Brown PO, Ginsberg MH: **Physical mapping of genes in somatic cell radiation hybrids by comparative genomic hybridization to cDNA microarrays.** *Genome Biol* 2002, **3**:RESEARCH0026.
 10. Baerends RJ, Smits VVK, De Jong A, Hamoen LW, Kok J, Kuipers OP: **Genome2D: a visualization tool for the rapid analysis of bacterial transcriptome data.** *Genome Biol* 2004, **5**:R37.
 11. Chi B, DeLeeuw RJ, Coe BP, MacAulay C, Lam WL: **SeeGH - A software tool for visualization of whole genome array comparative genomic hybridization data.** *BMC Bioinformatics* 2004, **5**:13.
 12. Greshock J, Naylor TL, Margolin A, Diskin S, Cleaver SH, Futreal PA, deJong PJ, Zhao S, Liebman M, Weber BL: **1-Mb resolution array-based comparative genomic hybridization using a BAC clone set optimized for cancer gene analysis.** *Genome Res* 2004, **14**:179-187.
 13. **Java** [<http://java.sun.com/>]
 14. **Java Web Start** [<http://java.sun.com/products/javawebstart/>]
 15. **Caryoscope** [<http://caryoscope.stanford.edu/>]
 16. **General Feature Format (GFF)** [<http://www.sanger.ac.uk/Software/formats/GFF/>]
 17. Diehn M, Sherlock G, Binkley G, Jin H, Matese JC, Hernandez-Bousard T, Rees CA, Cherry JM, Botstein D, Brown PO, Alizadeh AA: **SOURCE: a unified genomic resource of functional annotations, ontologies, and gene expression data.** *Nucleic Acids Res* 2003, **31**:219-223.
 18. **Open Source Initiative (OSI)** [<http://www.opensource.org/>]
 19. **The MIT License** [<http://www.opensource.org/licenses/mit-license.php>]
 20. **Aliasing Problems and Anti-Aliasing Techniques** [<http://www.siggraph.org/education/materials/HyperGraph/aliasing/alias0.htm>]
 21. **Digital Imaging and Communications in Medicine (DICOM)** [<http://medical.nema.org/>]
 22. **CGHAnalyzer** [<http://guanine.genomics.upenn.edu/people/faculty/weberb/CGH/html/>]
 23. **SeeGH** [<http://www.bccrc.ca/ArrayCGH/>]

Publish with **BioMed Central** and every scientist can read your work free of charge

"BioMed Central will be the most significant development for disseminating the results of biomedical research in our lifetime."

Sir Paul Nurse, Cancer Research UK

Your research papers will be:

- available free of charge to the entire biomedical community
- peer reviewed and published immediately upon acceptance
- cited in PubMed and archived on PubMed Central
- yours — you keep the copyright

Submit your manuscript here:
http://www.biomedcentral.com/info/publishing_adv.asp



BioMed Central publishes under the *Creative Commons Attribution License (CCAL)*. Under the *CCAL*, authors retain copyright to the article but users are allowed to download, reprint, distribute and /or copy articles in BioMed Central journals, as long as the original work is properly cited.