Protein Array Analyzer for ImageJ

Gilles Carpentier[a] and Emilie Henault[a]

[a]Faculty of Sciences and Technology - University Paris-Est-Creteil Val de Marne, France.

ABSTRACT

Protein array is used to check semi-quantitatively the presence of a specific protein interaction inside a sample. The *Protein Array Analyzer*, which was programmed in ImageJs macro language, is an extention of the *Dot Blot Analyzer*,[1] a graphically interfaced tool which greatly simplifying analysis. Once opened, each protein array image is integrated into its own graphical interface for density analysis. Single analysis can be then masterized, giving normalized and modellized representations with data table.

Keywords: Protein array, cytokine antibody array, dot blot, macro language, graphical interface

1. INTRODUCTION

Protein array is used to analyze protein expressions by screening simultaneously several protein-molecule interactions such as protein-protein and protein-DNA interactions. In most cases, the detection of interactions leads to an image containing numerous lines of spots that will be analyzed by comparing tables of intensity values. To describe the observed different patterns of expression, users generally show histograms with the original associated images.[2] The *Protein Array Analyzer*[3] gives a friendly way to exploit this type of analysis, thus allowing quantification, image modelling and comparative analysis of patterns.

2. MATERIALS AND METHODS

Among the large number of available protein array techniques, the data exhibited here for demonstration were obtained from mouse cytokine antibody arrays.[4] 62 distinct antibodies are arrayed, along a 14 x 10 grid displayed on 26 x 17 mm membranes. Biological samples have been obtained from plasma of mice treated with a drug possibly modulating cytokin levels. Chemiluminescence detection was imaged after x-ray film exposition, using a CCD camera device. The software was programmed in ImageJs macro language and was graphically interfaced, which greatly simplifies analysis. This project is the continuation of a previous work, the *Dot Blot Analyzer*[1] that already permits analysis of spot arrays through a graphical interface.

3. FEATURES

3.1 First step: analysis of single arrays

Quantitation of protein array images consists in integrating the pixel values surrounded by a user-defined circular selection. To help the user, a cursor gives dynamically the coordinates and the intergrated value of the target area. Once an analysis has started by the *Array Analysis* menu, the analyzer builds a preview of the array containing some sensitive areas allowing numerous settings: users can optimize the visual aspect of the array area, by adjusting lut and histogram parameters without altering the measurements. The cursor is also customizable by changing size and color. The graphical interface allows the *grid mode* analysis parameters' settings. This *grid mode* allows to define a matrix of circles to be measured, by selecting three dots forming the left/vertical axis angle. The number of lines and rows of the grid can be adjusted. This permits to define several analysis areas into the same array image. This mode also offers as option, an array modellized image, calculated from the measured values, which improves the display, see Fig. 1.

Send correspondence to Gilles Carpentier. E-mail: carpentier@u-pec.fr

Carpentier G and Henault E. Protein Array Analyzer for ImageJ. 2010. 3nd ImageJ User and Developer Conference proceedings. ISBN 2-919941- 11-9 : 238-240.

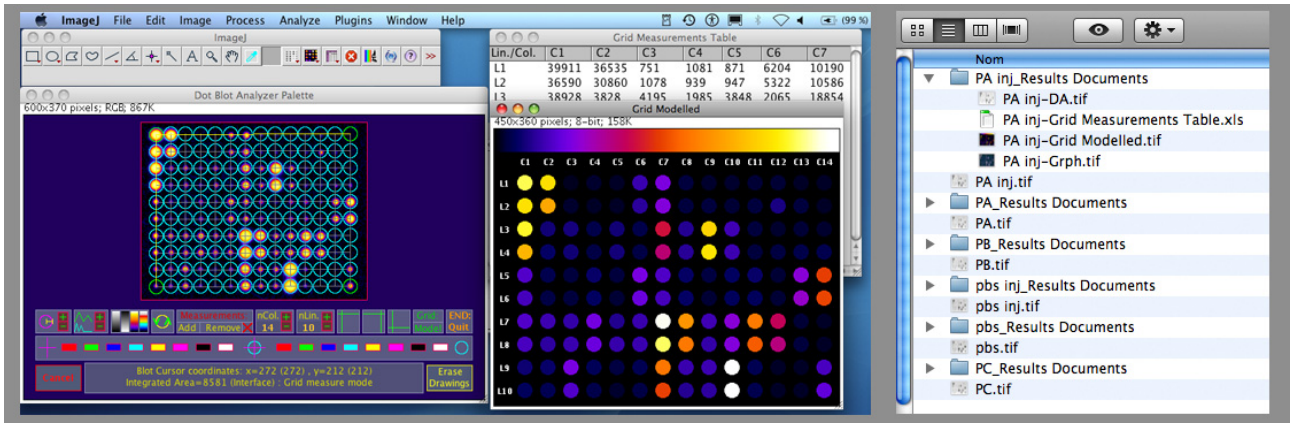

Figure 1. Each array is analysed using the utilities of the graphical interface (left). Result documents are saved on the hard disk as folders (right). The master pattern will be created by reading *Grid Modelled* tables contained in this kind of folder.

3.2 Second step: masterize single array analysis

Once arrays are individually analyzed using the *Array Analysis* tool, the *Group Pattern* tool then allows to obtain a global view of a set of arrays (Fig. 2). The *Masterize from Analysis Repertories* function looks for result tables comming from the *Array Analysis* function, in the parent folder chosen by the user. The tool explores any sub-levels, and builds a master image, or pattern, associated to a master table prensenting all the found results. It is so possible to obtain a global panel of all the arrays of a set of experiments by a simple click. The default master pattern presents the arrays as they came from the analysis, with the visualization scaled between the minimum and the maximum values encountered in each array. Among the three other representations available, the *normalize according the min/max of the master* option gives the most natural aspect of the modelled pattern compared to the initial images. It can be used to estimate visually the global intensity of an array among the group. The two last representations concern the normalization of the master pattern. In the array analyzed here, the manufacturer has provided some control dots and positive internal references. The tool allows to enter 4 controls and 4 references, to optimize and normalize the detected values. It is possible to use the references as global normalization of the master, or for normalization of each array of the master. The figure 2 gives an example of the last one, which permits to discriminate the highest level of the array number 5. The user has other presentation facilities, such as the choice of the number of columns in the master pattern, visualize or suppress the outline of the dots and of the normalization references.

3.3 Internet ressources

The tool contains the generally required online functionalities of software: documentation, demo images for training and update facilities.

4. CONCLUSION

The presented work is an easy and flexible tool to analyze a single or a group of arrays of dots. The different methods of normalization provided permit an easy global analysis of a set of experiments, and a semi-quantitive ready to use data tables, and graphical representations

ACKNOWLEDGMENTS

Special thanks to Angelica Keller and Alessandra Albano for their critical readings.

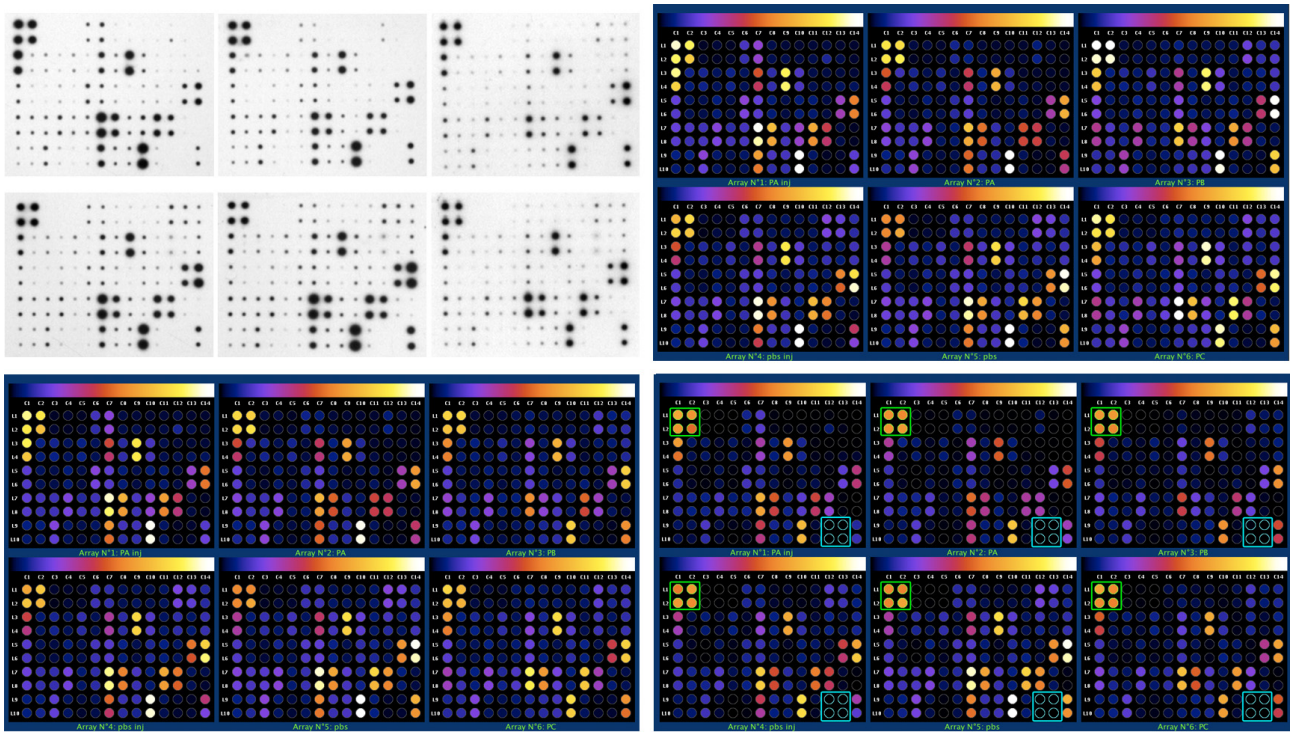

Figure 2. Initial images of the 6 protein arrays provided as demonstration for training (high-left). Masterized pattern using the initial normalization method: the model of each array is optimized from its minimum and maximum values, (high-right). Master pattern using the global normalization from the minimum and maximum values found in the master: this representation is the closest to the natural initial images (bottom-left). Master pattern obtained using internal normalization: mean of controls (surrounded and framed in cyan) was subtracted, and every value was normalized from the mean of positive references (surrounded and framed in green). The last representation allows to compare the level of one spot with all arrays. Note that the array 5 presents the highest level.

REFERENCES

[1] G. Carpentier, "Dot blot analyzer: Software development using the macro language of imagej," in *Proceedings of the ImageJ User and Developer Conference*, C. de Recherche Public Henri Tudor, ed., (ISBN 2-919941-06-2), pp. 3–5–189, 2008.

[2] M. Watanabe, W. Guo, S. Zou, S. Sugiyo, R. Dubner, and K. Ren, "Antibody array analysis of peripheral and blood cytokine levels in rats after masseter inflammation," *Neuroscience Letters* (382), pp. 128–133, 2005.

[3] G. Carpentier, *Protein Array Analyzer for ImageJ - Documentation.* http://image.bio.methods.free.fr/ImageJ/?Protein-Array-Analyzer-for-ImageJ.html, 2010.

[4] RayBio, *Mouse Cytokine Antibody Array Protocol (manual).* http://www.raybiotech.com/manual/mouse-manual.pdf, 2009.