

Data collection at the Edinburgh Mouse Atlas

Richard Thomas, 15 July 2003
richard@csse.uwa.edu.au

Introduction

This note discusses some of the issues faced by the scientific editors in the Edinburgh Mouse Atlas Project (EMAP). The aim is to examine the potential to capture user actions and track common tasks so that they can be analysed for usability improvements.

The EMAP project is led by Richard Baldock and Duncan Davidson of the UK MRC Human Genetics Unit. Information was gathered mainly during two fact finding visits in June 2003. A list of resources is given in the next section.

Data on gene expression in mouse embryos is organised into a textual plus 2D and 3D image database (EMAGE) which is housed in a common spatial and textual framework (The Mouse Atlas). A 3D rendering of gene expression is obtained through integration of a series of 2D sections, rather like an MRI.

In order to obtain information over the life of an embryo, data from each gene assay is classified according to the developmental stage of the embryo sample. The underlying scheme used in EMAP is Theiler Staging [Theiler 1989].

There are a variety of assay techniques. In EMAP all data is obtained from outside laboratories, and currently most incoming data is from the GXD [Hill et al 2004] database of the Jackson Labs in the USA. The main transmission is currently through papers which contain a textual description on the gene expression with an accompanying image. Scientific editors in EMAP insert the text and image data into EMAP in a process requiring scientific knowledge, skill at the task and accuracy. There are currently three editors and they can expect to process several hundred papers in a few months.

Resources

The project seems to be very well documented and organised. Apart from the main website there is a CDROM with some software and also supporting documentation.

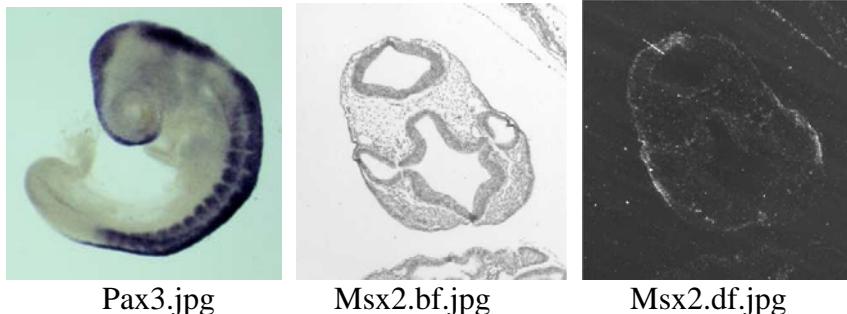
- Main website at <http://genex.hgu.mrc.ac.uk/intro.html>
 - EMAGE database
<http://genex.hgu.mrc.ac.uk/Emage/database/intro.html>
 - Mouse Atlas <http://genex.hgu.mrc.ac.uk/Atlas/intro.html>
- The CD-ROM has a copy of the Mouse Atlas and MAPaint software

- UK HGMP Resource Centre & EMAP. *Introduction to the EMAP 3D digital atlas of mouse development and EMAGE gene expression database*. Training Course Notes, held at Dept of Genetics, University of Cambridge, May 28-30 2003. (contains detailed examples of how to perform tasks)
- Manual of *Standard Operating Procedures* for the EMAP editors
- Images I was given for a talk, available at <http://www.csse.uwa.edu.au/~richard/emap/>
 - Screenshots about the process are tie-point1.jpg, tie-point2.jpg, threshold.jpg, text_annotation.jpg
 - Sample embryo files are Pax3.jpg, Msx2.bf.jpg, Msx2.df.jpg, AP2.jpg

Workflow

The process of taking a paper from, say, GXD and curating all the data is as follows:

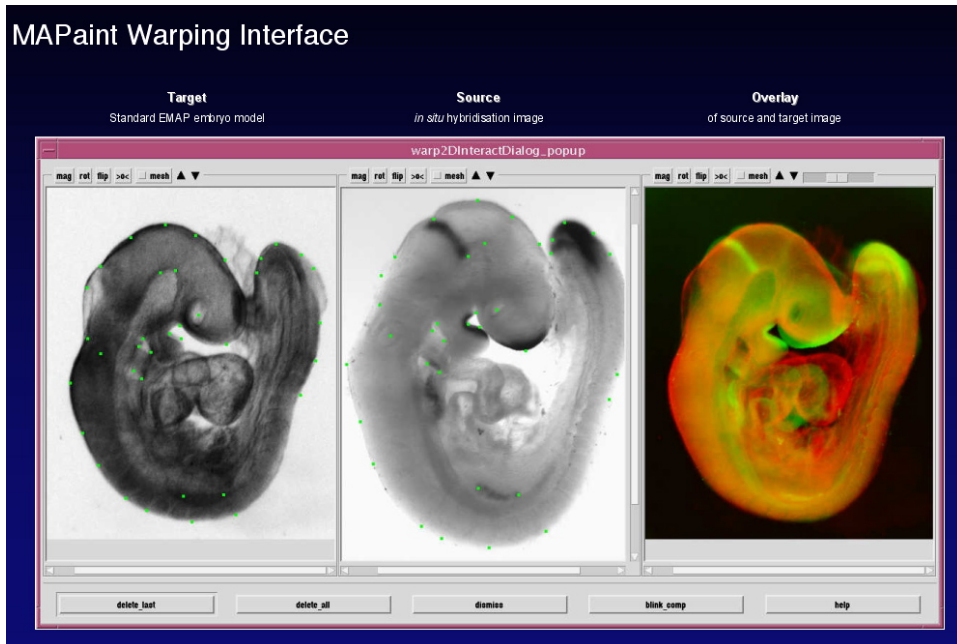
1. Search GXD database for all relevant entries of one gene
2. Read paper from where this data came, understand what is expressed and where, and identify the Theiler stage (as this is extremely unlikely to be used by authors, the more common and less precise days *post coitum* value is most often used).
3. Start to build database entries in both the in-house 'GXD-screened' Axiope database and the in-house AdminTool interface (which will make an entry in the EMAGE database).
4. Save, and possibly crop, a copy of each image from the GXD entry in a separate file in a directory for that paper. Examples of cropped images are



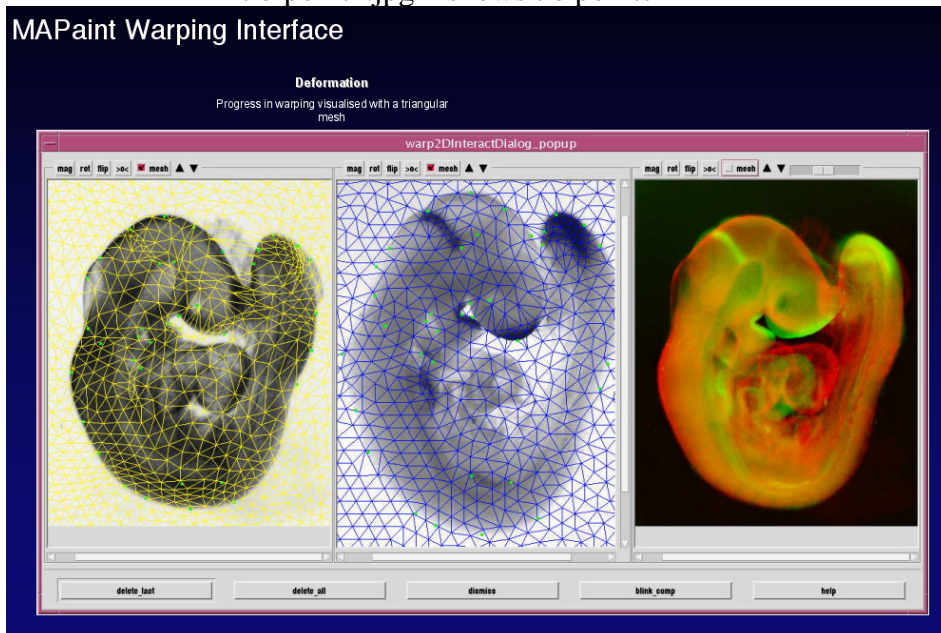
5. Convert the images to WLZ format (a Windows disk image archiving format), reducing any colour information to greyscale.
6. Try to stage the embryo from the anatomical details visible in sections from the database and supplied images. Different stages can be seen on the Atlas homepage <http://genex.hgu.mrc.ac.uk/Atlas/intro.html> or 3D interactive models can be read from the CD-ROM. The stages are based on Theiler staging and require detailed understanding of developmental anatomy, substages and variations. If the sample embryo image does not fit the Atlas model stages, a database entry is created but the data is not spatially mapped. An additional complication for section data is to determine the

section within the correctly staged embryo model which is constructed from a series of planar sections.

- Warp the input data image to fit the layout of the embryo in the database. This involves creating tie-points between source and target images. The process is about to be augmented by a program that provides a partial solution for editing. The following two examples illustrate this:



tie-point1.jpg – shows tie points



tie-point2.jpg – shows subsequent warping

8. Set a threshold for the warped image so that the gene expression shows up in the anatomical regions of the embryo model. Sometimes there will be multiple thresholds to represent. Thresholding is a highly variable and subjective task. Sometimes the aim is to feature outer surfaces or walls of cavities, while at other times internal regions are the main concern. The MAPaint interface appears as follows:



threshold.jpg

The rate of image curation is not known, but is believed to be 30-300 per month.

Systems

All the work of the editorial office is supported by a Sun server running Solaris. Editors use *tssh* and X.

There is an AXIOPE database (<http://www.axiope.org/>) which is used to track which images from GXD have been assessed for entry into EMAGE by editorial staff. Once an image is deemed to be appropriate for inclusion in EMAGE, the AdminTool interface is used to create an EMAGE entry and is then used to track the progress of each editing job, from editing, quality control and release into the public database. The same database that is launched by the AdminTool in-house supports the public EMAGE database.

There is a good directory structure to store raw images, processed images, curation files and the like. Within each gene directory there are images whose file extension represents the processes performed to get to that image, e.g. *4a.r.wlz* means the red channel of figure 4a in the paper. It is procedurally important to ensure that these

files are in the correct directory. Backups occur every 24 hours, but complete earlier versions are not stored in the gene folders.

There is a series of utilities which convert raw images to WLZ format.

The main image processing takes place within the MAPaint tool. The tie-point and thresholding images above are screenshots of MAPaint. While thresholding, the editor is likely to examine several colour (signal) selections of the image as the each signal conveys specific information.

MAPaint is written in C and runs under Solaris. I understand that in about a year a new version will be completed. This will have Java interfaces to the C so that it can be run on Windows and Macs. The main benefit will be that other labs will be able to do their own editing and submit results to EMAP for curation and publication more easily.

An enhanced tie-point program is being developed that will partially do the job for the editor to complete. This is likely to take some months.

In a few months the intention is to use colour images (as opposed to greyscale) in MAPaint. This will substantially change the thresholding task or even eliminate it if a gradient of colour intensity is implemented. However some sort of equalisation of the gene expression pattern is likely to be required.

With the current system it would be possible to extract all the raw images and also the final thresholded results. Thus a substantial set of raw and finished image pairs is already available for subsequent analysis.

The organisation of the workflow suggests that a large amount of the activity of the editorial office could be captured from shell history and database entries. Furthermore with some other tool it might be possible to track all the actions within MAPaint. (WOSIT – <http://www.openchannelfoundation.org/projects/WOSIT/> – might be a starting point for a UAR clone.)

Importantly, the Atlas and database are available to outside users over the web. Java is used extensively for this.

Data analysis

There is a range of possibilities to detect the procedures/sequences of editors and other users. The following are speculative, but many have been mentioned to me:

- Track who uses the Atlas, what they do etc. Could help with optimisation, workload planning, deciding where to extend the detail and so on.
- Version and configuration control

- Quality control
- Perhaps a tool, as a memory aid, to rebuild the context of what someone did months ago and now has to revisit
- Tools to support the virtual community of laboratories (e.g. Jackson), editors and Atlas users
- Setting thresholds for gene expression patterns
- Marking tie-points
- Extraction of site of expression from text
- Distribute some tasks to less skilled people
- Identify common work flows based on actual usage profiles rather than task analysis

Machine Learning

Initially the thresholding task looked very promising for machine learning. There is already a good training set of examples. Furthermore it might be possible to capture the process of setting the threshold and therefore see what the editor rejected (track the slider). However this task may become obsolete once the use of colour images is in place.

Any machine learning would need to be performed by a specialist. Amos Storkey of the University of Edinburgh's School of Informatics has discussed this with me and makes several caveats. It is important the task is significant, is an "interesting" problem academically and fits the profile of a research group. Also thresholding might be a task that can be performed by more conventional means. Lastly the actual work would perhaps be done as an MSc project under his direction. These start in May of each year, have an uncertain outcome and so on. It is not clear that the task is suitable for an MSc either, as some examples seem very hard, requiring much anatomical knowledge and judgement.

Discussion

There seems to be great potential to capture user actions and track common tasks. The GRUMPS project at Glasgow (<http://grumps.dcs.gla.ac.uk>) addresses how to do this generically. In principle there might be some common ground between GRUMPS and EMAP. However some caveats need to be mentioned.

There are three stages in the collection process. First, instruments must be created to trap user actions. So far the GRUMPS UAR tool only does this for the Windows platform, so re-engineering would be required. Also other instruments might be required to extract data from command histories and directories. Another instrument would have to be built for any web monitoring. A rough estimate is 1 person-month per instrument, perhaps a bit less.

Second, the means of data transmission from instruments to repository of user actions will have to be created and maintained. The repository design would likely be the same as in the current GRUMPS experiments as it has proved robust and effective. However some mechanism should be implemented to undertake periodic cleaning of the repository.

Third, time needs to be allowed for data cleaning and preparation. From my own experience and the student projects at Glasgow and UWA, a budget of one person month would be a good starting point. However the task might take 2 weeks or 2 months.

All the above activity eventually leads to a data product that could be used for machine learning or some other means of system development. In my view there should be agreed plans for this before any work begins. Amos Storkey might be interested to use such a product. Alternatively, the EMAP project programmers themselves could benefit. *It is essential this be worked out beforehand.*

Lastly, there are privacy issues. The EMAP users over the web may be concerned that the focus of their work can be inferred from the pattern of queries. In-house, any monitoring of editor activity could have privacy implications and might require ethics committee approval as personal data would be captured.

Conclusion

There seems to be potential to use GRUMPS-like technologies to generate a data product of user actions in the EMAP project. However a clear agenda needs to be established first.

References

Theiler, K "The House Mouse: Atlas of Mouse Development" Springer-Verlag, New York, 1989

Hill DP, Begley DA, Finger JH, Hayamizu TF, McCright IJ, Smith CM, Beal JS, Corbani LE, Blake JA, Eppig JT, Kadin JA, Richardson JE, Ringwald M. The mouse Gene Expression Database (GXD): updates and enhancements. Nucleic Acids Res. 2004 Jan 1;32 Database issue:D568-71.

Acknowledgements

This work was partially supported by the National eScience Centre, Edinburgh. The EMAP team is gratefully thanked, especially Jeff Christiansen, who commented on an earlier draft, and Richard Baldock. Thanks also to Amos Storkey, Malcolm Atkinson and Phil Gray for their discussions.

