Matching Spatial Regions with Combinations of Interacting Gene Expression Patterns

Jano van Hemert[1] and Richard Baldock[2]

[1] National e-Science Centre, School of Informatics, University of Edinburgh, UK
J.vanHemert@ed.ac.uk
[2] Human Genetics Unit, Medical Research Council, Edinburgh, UK
Richard.Baldock@hgu.mrc.ac.uk

Abstract. The Edinburgh Mouse Atlas aims to capture *in-situ* gene expression patterns in a common spatial framework. In this study, we construct a grammar to define spatial regions by combinations of these patterns. Combinations are formed by applying operators to curated gene expression patterns from the atlas, thereby resembling gene interactions in a spatial context. The space of combinations is searched using an evolutionary algorithm with the objective of finding the best match to a given target pattern. We evaluate the method by testing its robustness and the statistical significance of the results it finds.

Key words: gene expression patterns, in situ hybridization, spatio-temporal atlases; evolutionary algorithms

1 Introduction

The location of expressing genes can be revealed by performing *in-situ hybridisation* on either embryo sections or wholemount embryos. This process uses labelled RNA that binds to mRNA in the cell, which is a good indication the corresponding gene in the cell is active. Essentially, the result is a stained embryo or part thereof, where the stain indicates where the gene is expressing. As these patterns exhibit gradients, and as the process is quite sensitive, the resulting data needs careful examination before inferring the location of a gene expression pattern. The Edinburgh Mouse Atlas Project (EMAP) [1] has a curators office, which performs these examinations and translates these patterns into the common spatio-temporal framework for the developing *Mus Musculus*.

EMAP is a unique resource that captures data in one common spatio-temporal framework, thereby opening the possibilities to perform queries and analyses in both embryo space and embryo development time to explore how genes interact on an inter-cellular level. Currently the spatial data can be queried by defining a pattern in the context of the embryo. The database will then return all gene expression patterns that intersect with the query domain, sorted by similarity with that domain. In addition the spatial patterns can be clustered in terms of spatial similarity to reveal putative syn-expression groups. More sophisticated analysis involving pattern combinations is not however possible. Recently the push is towards gene marker studies that try to redefine the spatial context of embryos in terms of where genes are expressing.

We have developed a methodology to define a given target pattern by the combination of multiple gene expression patterns via several gene interactions operations. The target pattern can be the expression pattern of a particular gene, a pattern defined by a human, a pattern defined by anatomical components or by any other means of defining spatial area within the context of the model mouse embryo. The method searches for a set of genes and combines their expression patterns using predefined operations to closely match the target pattern, thereby attempting to define this pattern spatially. The objective of this study is to measure the robustness of the methodology and validate the significance of the resulting gene interactions. This is important as much noise exists in the acquisition of the patterns as well as much inaccuracy may exist in the target pattern.

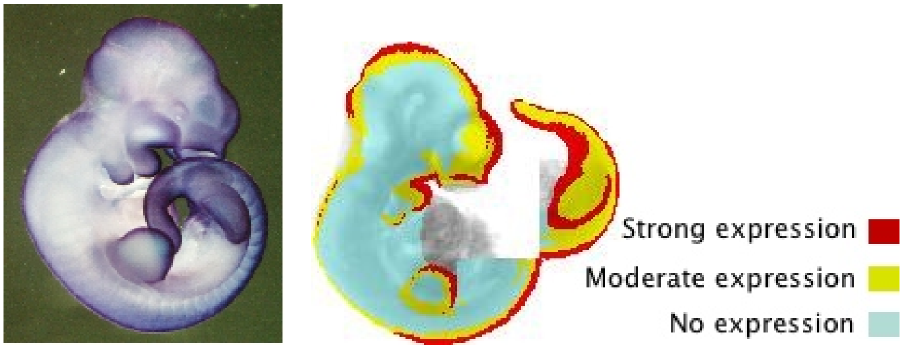
In the next section we describe the Edinburgh Mouse Atlas Project, a spatio-temporal framework for capturing anatomy and gene expression patterns in developing stages of the mouse. Then, in Section 3, we describe the methodology for constructing and searching gene interaction trees. Experiments and results are provided in Section 4. Last, we provide a discussion in Section 5.

2 Edinburgh Mouse Atlas Project

EMAGE (http://genex.hgu.mrc.ac.uk/) is a freely available, curated database of gene expression patterns generated by *in situ* techniques in the developing mouse embryo [1]. It is unique in that it contains standardized spatial representations of the regions of gene expression for each gene, denoted against a set of virtual reference embryo models. As such, the data can be interrogated in a novel and abstract manner by using space to define a query. Accompanying the spatial representations of gene expression patterns are text descriptions of the sites of expression, which also allows searching of the data by more conventional text-based methods terms.

Data is entered into the database by curators that determine the level of expression in each *in situ* hybridization experiment considered and then map those levels on to a standard embryo model. An example of such a mapping is given in Figure 1. The strength of gene expression patterns are classified either as no expression, weak expression, moderate expression, strong expression, or possible detection. Possible detection means the curator is uncertain whether the region exhibits gene expression, hence we exclude these regions from our analyses.

In this study we restrict to a subset of the data contained in the database. This subset of data originates from one study [2] and contains 1970 images of *in situ* gene expression patterns in a wholemount developing mouse embryo model of Theiler Stages 15–19 [3]. The study includes 1131 genes; a subset of genes were screened two or three times. By mapping the strong and moderate expression patterns of these images on to the two-dimensional model for Theiler Stage 17 shown in Figure 1(b), we can work with all these patterns at the same time.



(a) Original image shows expression of Hmgb1 in a mouse embryo at Theiler Stage 17

(b) Standard embryo with mapped levels of expression

Fig. 1. An example of curating data; the gene expression in the original image on the left is mapped on to the standard embryo model of equal developmental stage on the right (entry EMAGE:3052 in the online database)

3 Combining Gene Expression Patterns

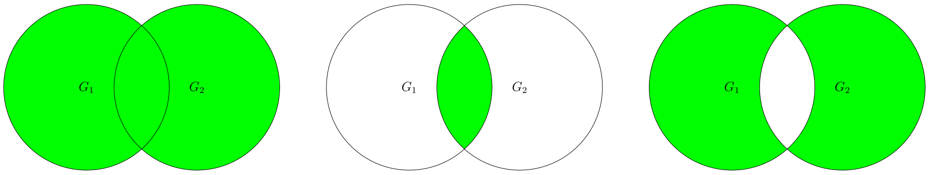
To make possible a directed search for a given target pattern we need to define how patterns can interact, to define how interactions are structured, to define a function that allows to measure the quality of matching two patterns and to have a mechanism whereby we can search the space of pattern interactions. The following sections will discuss each of these in detail.

3.1 Defining the Interaction Patterns

The interaction patterns are shown in Figure 2. Each interaction pattern is an operation on a spatial pattern where the operation, either OR, AND or XOR, is performed over each pixel[3] in the space as defined by the pattern. The AND operation is also referred to as the conjunction of patterns, whole the OR operation is referred to as the disjunction of patterns. This definition on sets takes every pixel location as an item and then a pattern is defined as a a set of pixel locations.

In terms of gene interactions, the AND operation represents two genes that require co-location in time and space in order to express. The XOR operation represents two genes cancelling each other's expression out when co-located. The OR operation merely takes the conjunction and no visible interaction occurs.

[3] Alternatively in a dimension higher than two it will be performed over a voxel



(a) OR: conjunction (b) AND: disjunction (c) XOR: conjunction except for the disjunction

Fig. 2. Types of interactions of two gene expression patterns G_1 and G_2

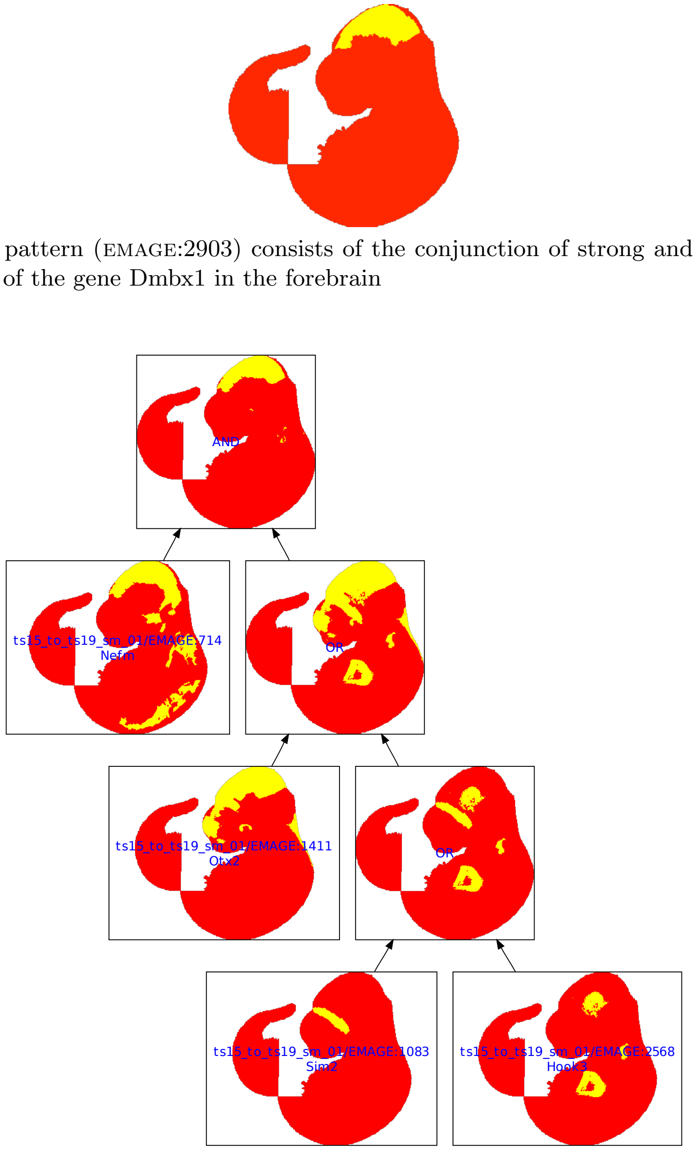
3.2 A Grammar for Interactions

Below, we define a simple BackusNaur form grammar [4] to allow arbitrary large interaction trees to be constructed. In practice, the size of these trees is restricted in the search algorithm. The operations are the binary operations over patterns as defined in the previous section and the existing patterns are unique identifiers to studies in the Edinburgh Mouse Atlas database (http://genex.hgu.mrc.ac.uk/), where in the study we have used the conjunction of strong and moderate strength gene expression patterns.

```
<pattern> ::= <existing pattern> | (<pattern> <operation> <pattern>)
<operation> ::=  AND | OR | XOR
<existing pattern> ::= EMAGE:1, EMAGE:2, ..., EMAGE:1970
```

When combining patterns in this fashion it is possible to create the empty set. For example, let us apply AND to two patterns, where one pattern expresses only in the tail and the other pattern only expresses in the head; as these patterns do not overlap, the operation will yield an empty pattern. This has two major consequences. First, sub-trees that produce empty patterns can increase the complexity of a large tree without adding any value. Second, allowing useless sub-trees inflates the size of the search space. To counteract this, we prune these empty patterns from trees by removing their sub-trees. If exactly one of the inputs of an operation is empty, we replace the operation by the non-empty pattern. If both patterns are empty, the parent will resort to an empty pattern itself and then let the next parent node deal with the problem. As this is a recursive mechanism. it does mean the result of a whole tree can be empty if the root node produces an empty pattern, in which case we discard the whole tree from the search. This procedure has two positive effects. First, the search space is reduced as useless sub-trees are not considered. Second, in the experiments this has lead to the prevention of so-called bloat [5, page 191] in the evolutionary algorithm, which is described later.

Worth noting is that the operation NOT cannot be included in the operations due to the nature of the patterns. A gene expression pattern is the result of a stain observed through a curation process. If we would negate the pattern, i.e., take the whole embryo and subtract the pattern, we then explicitly assert that


AND
ts15_to_ts19_sm_01/EMAGE:714
Nefm
OR
ts15_to_ts19_sm_01/EMAGE:1411
Otx2
OR
ts15_to_ts19_sm_01/EMAGE:1083
Sim2
ts15_to_ts19_sm_01/EMAGE:2568
Hook3

(a) Target pattern (EMAGE:2903) consists of the conjunction of strong and moderate expression of the gene Dmbx1 in the forebrain

(b) Evolved interaction tree, that tries to match the target pattern in Figure 3(a) and conforms to the grammar: (EMAGE:714 AND (EMAGE:1411 OR (EMAGE:1083 OR EMAGE:2568))); the similarity to the target is equal to 0.804

Fig. 3. An example of an interaction tree and a target. The yellow parts (bright) represent strong and moderate expression of the corresponding genes in the model embryo, represented in red (darker)

no gene expression occurs in the negated pattern. This assertion is false on many grounds; the curator may have seen only part of the embryo and the curator also notes weak and possible expression, which is not considered in the patterns we use here.

3.3 Matching Patterns

A function is required to compute the quality of a match between two patterns. This will allow the evolutionary algorithm described next to direct its search toward better matches. Given two patterns p_1 and p_2, we measure their similarity using the Jaccard Index [6]:

$$\text{similarity}(p_1, p_2) = \frac{\text{area}(p_1 \cap p_2)}{\text{area}(p_1 \cup p_2)}, \tag{1}$$

In Figure 3(a), a target pattern is given. The similarity of this pattern with the result from the evolved interaction tree, i.e., the pattern in the root of the tree, shown in Figure 3(b) is equal to 0.804.

The Jaccard Index was used in two previous studies on hierarchical clustering and association rules mining in which it gave the best results. It will be used also as a measure of quality of success in the experiments.

3.4 An Evolutionary Algorithm to Search for Sentences

The evolutionary algorithm [7] operates on the representation defined in Section 3.2, i.e., a binary tree where each internal node is one of the three binary operators defined over images and each leaf is a pattern takes from a predefined set of patterns.

Initially one hundred trees are randomly generated by growing them randomly [8]. A stochastic process is used to determine at each decision point whether a given node becomes either a leaf node, i.e., a pattern, or an internal node, i.e., an operation. If it becomes an operation we repeat this process for all the children of that node. The stochastic process makes a node a leaf with the probability of $1/(1 + \text{depth of the node in the tree})$ and an internal node otherwise. The actual choice of operation or pattern is a random uniform selection. Important to note is that the same mechanism is used to create random trees that serve to provide the target patterns in the experiments.

To create new individuals, two genetic operators are used. A crossover which picks one node (which can be a leaf) uniform randomly in two distinct trees and then replaces the sub-tree of the node pointed at in the second tree with the sub-tree of the node pointed at in the first tree. The result is the new offspring, which then undergoes mutation by again selecting a node (which can be a leaf) and then replacing its sub-tree with a randomly generated sub-tree. Trees that exceed 200 nodes and leafs are discarded, although this has not occurred in the experiments.

The objective function is the similarity function (Equation 1), which needs to be maximised. The offspring will always replace the tree that represents the pattern with the worst match, i.e., lowest fitness, in the population.

The algorithm terminates when the mean fitness of the population has converged to a preset value. More specifically, if μ is the mean fitness of the population and σ is the standard deviation of the population, then the algorithm terminates if $\frac{\mu}{\mu+\sigma} \geq 0.85$. To ensure timely termination, a maximum is set of 5 000 evaluations. The algorithm also terminates if a perfect match with the target pattern is found, i.e., if the optimisation function is equal to 1.0.

4 Experiments and Results

We perform two experiments. The first experiment will be used to determine the robustness of the method. More specifically, it will be used to determine the number of runs required of the evolutionary algorithm to get a reliable result. The second experiment will take patterns from the gene expression database and use these as targets for the methodology, after which the significance of the interaction trees is validated using an overrepresentation analysis.

4.1 Robustness of the Methodology

To evaluate the robustness of the approach we devise an experiment whereby target patterns are randomly perturbed. We provide the perturbed pattern to the evolutionary algorithm and measure how well it is able to match the original pattern. Both the amount of perturbation and how well the evolutionary algorithm matches the original pattern are measured in the same way as the objective function of the evolutionary algorithm (see Equation 1).

The following procedure is repeated 261 times. We create a random tree in the same manner as described in the initialisation phase of the evolutionary algorithm. The resulting pattern of this tree forms the *original target pattern*. Each pixel in the original target pattern undergoes a translation using a uniform random distribution over a domain of -20 and $+20$ in both x and y directions; the size of the bounding box of the embryo domain is 267×258. This process yields a *perturbed pattern*. The evolutionary algorithm is then run sixty times[4] with a unique seed to its random generator with as its target pattern the perturbed pattern. The best solution of one run of the evolutionary algorithm is called the *output pattern*. On average the size a the tree creating the original target pattern consists of 6.30 nodes (with a standard deviation 3.90).

We measure the amount of perturbation of the original target pattern with the perturbed pattern using the similarity defined in Equation 1. We measure the success of a run of the evolutionary algorithm by applying the same function to the output pattern of the run with the original target pattern. If the similarity between these two patterns is 1.000 we then check if the evolved tree is equal to the tree that created the original target pattern.

[4] One run of the algorithm takes about six minutes on a 2.33Ghz Intel Core Duo

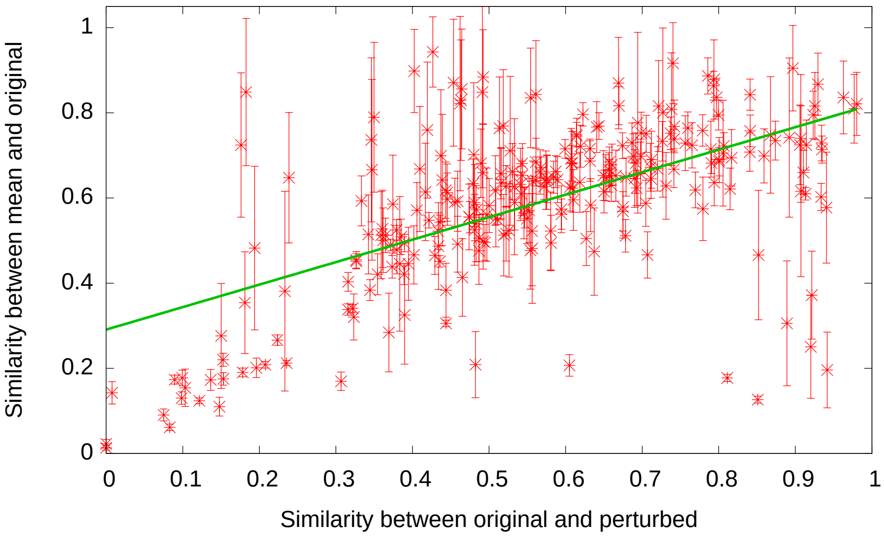


Fig. 4. Average amount of perturbation (*x-axis*) to the average success of matching the original pattern (*y-axis*). Every point is averaged over 60 unique runs of the evolutionary algorithm with 95% confidence intervals included. A linear regression is provided over the averaged points

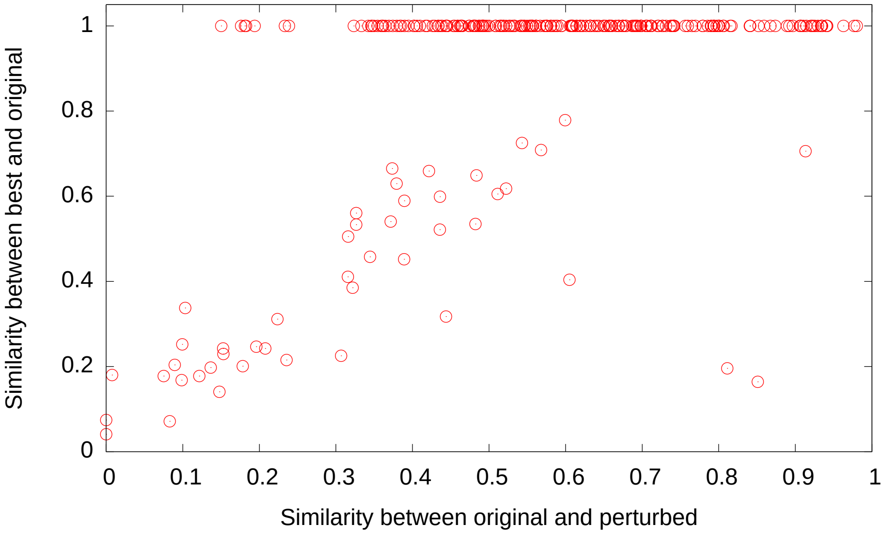


Fig. 5. Every point is the best match found (*y-axis*) in 60 unique runs of the evolutionary algorithm on one perturbed target pattern (*x-axis*)

Figure 4 shows the average success of the evolutionary algorithm in matching the original target pattern. Each point is the average of sixty unique runs and is accompanied by a 95% confidence interval. A regression is performed with Marquardt-Levenberg's nonlinear least-squares algorithm [9, page 683] on the averages, which results in a regression of $y = 0.528x + 0.291$ where the correlation coefficient is $r = 0.593$, the rms of residuals is 0.156, and the variance of the residuals is 0.024.

Figure 5 shows the best match found in the sixty runs for each original target pattern generated. The match is calculated between the best result from the evolutionary algorithm on the perturbed target pattern and the original target pattern. A large set of the points, 82.8% correspond to a similarity match of 1.0. After examining the corresponding trees for each of these matches, we confirm the evolutionary algorithm was able to recreate all the original trees for these cases.

4.2 Matching Gene Expression Patterns from the Mouse Atlas

We take the set of 1970 gene expression patterns as introduced in Section 2. Every pattern belongs to a gene for which the expression has been mapped to a standard model embryo. We repeat the following operation for every pattern in the set. The pattern is temporarily removed from the total set and is deemed the target pattern. The evolutionary algorithm will then try to construct a pattern that matches it as close as possible by creating an interaction tree that only makes use of the remaining gene expression patterns. In other words, we are using mapped gene expression patterns from the Mouse Atlas itself as target patterns.

For each target pattern we run the evolutionary algorithm sixty times, as the results from the experiments in Section 4.1 show this leads to robust solutions. The total number of runs of the evolutionary algorithm in this experiment becomes 118 200. For the resulting interaction tree of each run, we determine whether the genes used in that interaction tree are statistically overrepresented with respect to groups of genes associated with annotations in the *Gene Ontology* (GO) [10]. Numerous software package support this type of statistical verification. Here we use the free and Open Source GO::TermFinder Perl module [11]. It works by calculating a p-value using the following hypergeometric distribution without re-sampling:

$$p\text{-value} = \sum_{j=x}^{n} \frac{\binom{M}{j}\binom{N-M}{n-j}}{\binom{N}{n}} \tag{2}$$

In this equation, N is the total number of genes in the background distribution, which is all genes in our study and therefore equal to 1131, M is the number of genes within that distribution that are annotated (either directly or indirectly) to the node of interest in the Gene Ontology, n is the size of the list of genes of interest, i.e., in the tree interaction, and k is the number of genes within that list annotated to the node. To account for falsely finding significant hypothesis due to random chance given multiple events, we use Bonferroni correction. We deem results significant if $p < 0.05$.

The statistical analysis gives us a list of items, where each item consists of a target pattern that belongs to a gene, the resulting interaction tree of the corresponding run, a value to express the match between the target pattern and the pattern originating from the interaction tree as calculated using Equation 1, a p-value as calculated using Equation 2, a GO term G, and a list of genes that simultaneous appear in the interaction tree and are attributed to the GO term G. In addition to these results, we also include the size of the target pattern in relation to the total embryo. This helps us to discard patterns that take up almost all of the embryo, which tend to correspond to housekeeping genes.

Some disadvantages exist in using this methodology to validate statistical significance, as it depends fully on the current annotations of the Gene Ontology. It may prevent us from extracting new discoveries in the following ways. It may happen the evolutionary algorithm finds a set of genes that interact, but which are not associated with an annotated term in the Gene Ontology, or because the term simply does not exist. Another possibility is that the set of interaction genes is small and can be found by chance alone, in which case it will be discarded on the basis of the p-value. Also plausible is that genes cannot contribute to the likelihood that the interaction tree they are part of passes the significance test because although they do exist in the Mouse Atlas Database, they do not exist in the Gene Ontology,.

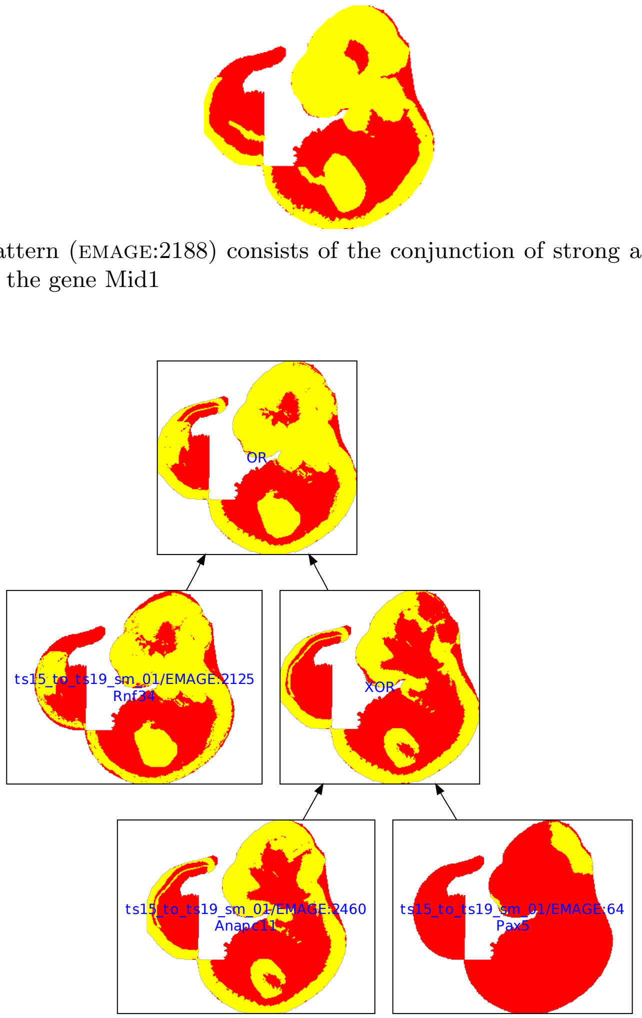
This overrepresentation analysis yields 6666 items from 1992 unique runs of the evolutionary algorithm. It is possible a set of genes is involved in multiple Gene Ontology terms. The total number of items is too large to include here or to ask a developmental biologist to poor over. We therefore filter the list of items by only including those where the result of the interaction tree matches with more than 0.70 similarity to the target pattern and the relative size of the target pattern is less than 0.50 of the embryo.

After filtering, we are left with 230 items, of which we present 45 filtering in Table 1. Each row in the table corresponds to the results of performing an overrepresentation analysis in the context of one GO term with the genes from one interaction tree that tries to match one gene expression pattern from the database. To illustrate the real output, we show the corresponding target pattern and the interaction tree of two of these results, which are shown in Figure 6 and Figure 7.

In Figure 7(b), the interaction tree shows all three types of interactions. First the intersection of the patterns of Hoxa9 and Gli3 is taken (AND), after which the result undergoes a XOR with the pattern of Otx2. Last, the result of that interaction is then merged (OR) with the pattern of the gene Lsr to form the result of the interaction tree. This result has a similarity of 0.722 when compared to the target gene expression pattern of Pygo1 shown in Figure 7(a) using Equation 1. In [12], the interaction between the genes Otx2 and Gli3 is described in the context of the development of the inner ear.

Table 1. Items resulting from checking statistically overrepresentation of genes against annotated Gene Ontology terms. Every row consists of the target pattern as an EMAGE id, the run number, the relative size of the target pattern, the match between the target pattern and the result from the interaction tree, the *p*-value from the overrepresentation analysis, the genes corresponding to both the tree and the GO term, and the GO term where the analysis was performed against.

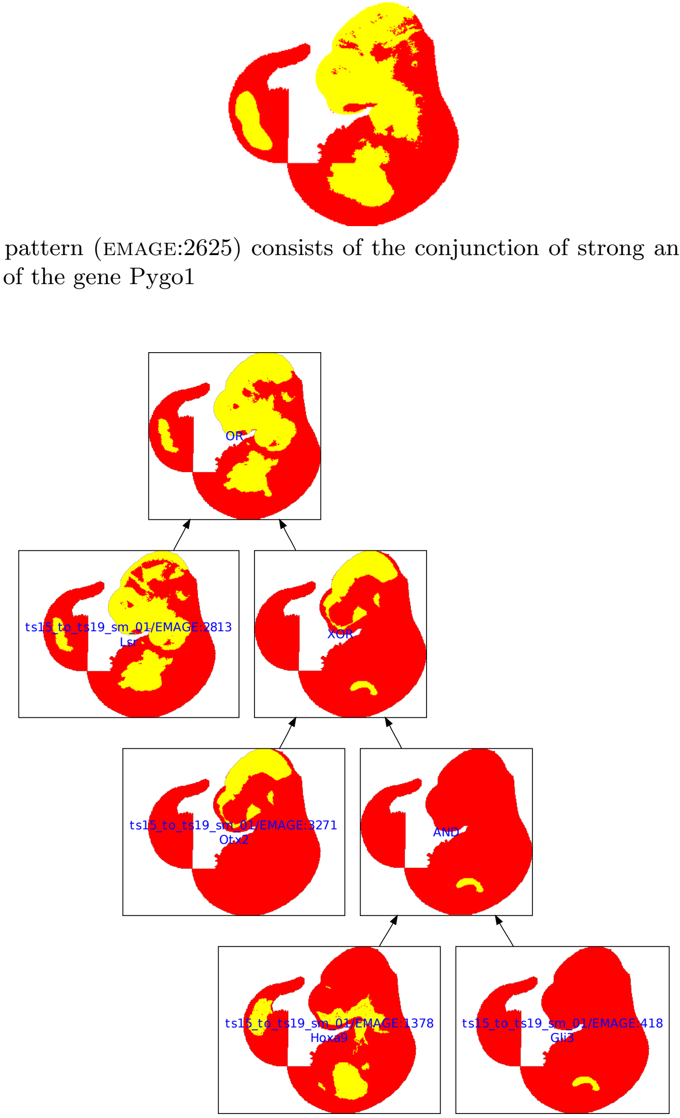
EMAGE	run	size	match	*p*-value	genes	GO term
1182	10	0.39	0.75	0.021	Ube2g2 Zfp36l2	cellular macromolecule catabolic process
1182	34	0.39	0.74	0.00066	Shh Pecam1	lipid raft
1182	34	0.39	0.74	0.0027	Ube2g2 Shh	proteasomal ubiquitin-dependent protein catabolic process
1182	34	0.39	0.74	0.027	Ube2g2 Shh	cellular protein catabolic process
1182	34	0.39	0.74	0.027	Ube2g2 Shh	modification-dependent macromolecule catabolic process
1182	34	0.39	0.74	0.027	Ube2g2 Shh	modification-dependent protein catabolic process
1182	34	0.39	0.74	0.027	Ube2g2 Shh	proteolysis involved in cellular protein catabolic process
1182	34	0.39	0.74	0.027	Ube2g2 Shh	ubiquitin-dependent protein catabolic process
1204	8	0.0049	0.72	0.0095	Lpp Cxadr Bcl6	biological adhesion
1204	8	0.0049	0.72	0.0095	Lpp Cxadr Bcl6	cell adhesion
1205	2	0.0041	0.74	0.029	Nkx2-2 Nkx6-2	pancreas development
1224	3	0.0059	0.71	0.007	Rnf6 Rnf14 Hlcs Pias2	ligase activity, forming carbon-nitrogen bonds
1224	3	0.0059	0.71	0.046	Rnf130 Rnf6 Mbtps1	proteolysis
130	18	0.024	0.73	0.0026	Actc1 Shh Nkx2-5 Bmp4	muscle cell differentiation
130	18	0.024	0.73	0.0029	Actc1 Shh Bmp4	myoblast differentiation
130	18	0.024	0.73	0.0081	Actc1 Shh Nkx2-5	striated muscle cell differentiation
130	18	0.024	0.73	0.012	Actc1 Shh Bmp4	muscle fiber development
130	18	0.024	0.73	0.012	Actc1 Shh Bmp4	skeletal muscle fiber development
130	18	0.024	0.73	0.013	Actc1 Shh Nkx2-5 Bmp4	striated muscle development
130	18	0.024	0.73	0.017	Nkx3-1 Shh	prostate gland development
130	18	0.024	0.73	0.017	Shh Bmp4	telencephalon regionalization
130	4	0.024	0.74	0.00013	Actc1 Nkx2-5 Bmp4	muscle cell differentiation
130	4	0.024	0.74	0.00045	Actc1 Nkx2-5 Bmp4	striated muscle development
130	4	0.024	0.74	0.0015	Actc1 Nkx2-5 Bmp4	muscle development
130	4	0.024	0.74	0.0033	Actc1 Bmp4	myoblast differentiation
130	4	0.024	0.74	0.0062	Actc1 Nkx2-5	striated muscle cell differentiation
130	4	0.024	0.74	0.008	Actc1 Bmp4	muscle fiber development
130	4	0.024	0.74	0.008	Actc1 Bmp4	skeletal muscle fiber development
130	4	0.024	0.74	0.023	Actc1 Bmp4	skeletal muscle development
130	7	0.024	0.73	0.02	Csrp3 Nkx2-5	cardiac muscle development
130	7	0.024	0.73	0.021	Actc1 Csrp3	I band
130	7	0.024	0.73	0.021	Actc1 Csrp3	contractile fiber
130	7	0.024	0.73	0.021	Actc1 Csrp3	myofibril
130	7	0.024	0.73	0.021	Actc1 Csrp3	sarcomere
1326	18	0.0049	0.71	0.047	Nr1i3 Nr2e3	steroid hormone receptor activity
1326	29	0.0049	0.74	0.021	Lpp Sox9 Tnxb Bcl6	biological adhesion
1326	29	0.0049	0.74	0.021	Lpp Sox9 Tnxb Bcl6	cell adhesion
1327	16	0.0059	0.74	0.033	Nefl Nbn	neuromuscular process controlling balance
1327	17	0.0059	0.76	0.025	Tnxb Zfp146	heparin binding
1327	17	0.0059	0.76	0.035	Tnxb Zfp146	carbohydrate binding
1327	17	0.0059	0.76	0.035	Tnxb Zfp146	glycosaminoglycan binding
1327	17	0.0059	0.76	0.035	Tnxb Zfp146	pattern binding
1327	17	0.0059	0.76	0.035	Tnxb Zfp146	polysaccharide binding
1327	29	0.0059	0.91	0.0041	Cdkn1c Mxi1 Nr2e3	negative regulation of transcription from RNA polymerase II promoter
1327	29	0.0059	0.91	0.009	Cdkn1c Mxi1 Nr2e3	negative regulation of transcription, DNA-dependent


OR
ts15_to_ts19_sm_01/EMAGE:2125
Rnf34
XOR
ts15_to_ts19_sm_01/EMAGE:2460
Anapc11
ts15_to_ts19_sm_01/EMAGE:64
Pax5

(a) Target pattern (EMAGE:2188) consists of the conjunction of strong and moderate expression of the gene Mid1

(b) Evolved interaction tree, that tries to match the target pattern in Figure 6(a) and conforms to the grammar: (EMAGE:2125 OR (EMAGE:2460 XOR EMAGE:64)); the similarity to the target is equal to 0.753

Fig. 6. An example of an interaction tree and a target. The yellow parts (bright) represent strong and moderate expression of the corresponding genes in the model embryo, represented in red (darker). The genes Anapc11, Rnf34 and Pax5 are overrepresented with respect to the GO terms *biopolymer modification*, *cellular macromolecule metabolic process*, *cellular protein metabolic process*, *post-translational protein modification* and *protein modification process* with p-values 0.0069, 0.021, 0.021, 0.0057 and 0.0064, respectively


OR
ts15_to_ts19_sm_01/EMAGE:2813
Lsr
XOR
ts15_to_ts19_sm_01/EMAGE:3271
Otx2
AND
ts15_to_ts19_sm_01/EMAGE:1378
Hoxa9
ts15_to_ts19_sm_01/EMAGE:418
Gli3

(a) Target pattern (EMAGE:2625) consists of the conjunction of strong and moderate expression of the gene Pygo1

(b) Evolved interaction tree, that tries to match the target pattern in Figure 7(a) and conforms to the grammar: (EMAGE:2813 OR (EMAGE:3271 XOR (EMAGE:1378 AND EMAGE:418))); the similarity to the target is equal to 0.722

Fig. 7. An example of an interaction tree and a target. The yellow parts (bright) represent strong and moderate expression of the corresponding genes in the model embryo, represented in red (darker). The genes Otx2 and Gli3 are overrepresented with respect to the GO term *cell fate specification* with p-value $= 0.046$

5 Discussion

With the increase of resolution and speed in which *in-situ* data can be captured [13,14], these type of studies become more useful in a similar manner to how microarray studies have become mainstream techniques. Where microarray gives insight into many genes at once, *in-situ* studies have the benefit of high precision in terms of spatial context. To make the data that result from from this technique more digestible to developmental biologists, we need mechanisms that allow investigation of multiple gene interactions within a spatio-temporal context. In this paper, we have identified, designed, implemented and evaluated one such mechanism in the spatial context a developing mouse embryo.

The approach uses a grammar to define a search space that allows several types of spatial gene interaction patterns to be combined. An evolutionary algorithm is used to search this space with the aim of maximising the match with a given target pattern. This target pattern can be created by a human, or represent a particular space of an embryo, such as anatomical components, the spatial expression pattern of one gene or even the combination of expression patterns of a number of genes. The output consists of interaction trees that show how a set of spatial expression patterns of multiple genes should be combined using either the conjunction, disjunction or exclusive OR operations over these patterns.

The domain of *in-situ* hybridization studies potentially contains much noise and uncertainties. To validate whether our approach will cope within such an environment we generated gene interactions that then represent target patterns. These patterns were then slightly perturbed to form a test suite that allowed us to analyse the robustness of the system. The results show that running the evolutionary algorithm sixty times and then selecting the best solution from those runs leads to sufficiently matching solutions. The results from the second experiment show how interaction trees can be evolved to match gene expression patterns in the same database. By using an overrepresentation analysis against annotated genes that correspond to terms in the Gene Ontology, we were able to show the method is able to extract statistically significant gene interactions.

Our future goal is to provide an interface freely accessible via any web browser to allow biologists to define target patterns that allow them to perform the combinatorial search introduced in this study.

Acknowledgements

This study is part of an EU-funded Design Study "Developmental Gene Expression Map" http://www.dgemap.org/, led by Susan Lindsay, which we thank for the fruitful discussions about spatial gene expression patterns. We acknowledge Bill Hill for his support in usage of the Woolz [15] libraries, wherein these data are made accessible, as well as the editorial team of the Edinburgh Mouse Atlas Project, Jeff Christiansen, Lorna Richardson and Shanmugasundaram Venkataraman, for curating the data from the Gray et al. [2] study. This work has made use of the resources provided by the Edinburgh Compute and Data Faclity (ECDF) http://www.ecdf.ed.ac.uk/.

References

1. Christiansen, J., Yang, Y., Venkataraman, S., Richardson, L., Stevenson, P., Burton, N., Baldock, R., Davidson, D.: Emage: a spatial database of gene expression patterns during mouse embryo development. Nucleic Acids Research **34** (2006) 637–641
2. Gray, P., et al.: Mouse brain organization revealed through direct genome-scale tf expression analysis. Science **306**(5705) (2004) 2255–2257
3. Theiler, K.: The House Mouse Atlas of Embryonic Development. Springer Verlag, New York (1989)
4. Knuth, D.: Backus normal form vs. backus naur form. Communications of the ACM **7**(12) (1964) 735–736
5. Langdon, W.B., Poli, R.: Foundations of Genetic Programming. Springer-Verlag (2002)
6. Jaccard, P.: The distribution of flora in the alpine zone. The New Phytologist **11**(2) (1912) 37–50
7. Bäck, T., Fogel, D., Michalewicz, Z., eds.: Handbook of Evolutionary Computation. Institute of Physics Publishing Ltd, Bristol and Oxford University Press, New York (1997)
8. Koza, J.: Genetic Programming: On the Programming of Computer by Means of Natural Selection. MIT Press (1992)
9. Press, W., Flannery, B., Teukolsky, S., Vetterling, W.: Numerical Recipes in C: The Art of Scientific Computing. 2nd edn. Cambridge University Press (1992)
10. Consortium, T.G.O.: Gene ontology: tool for the unification of biology. Nature Genet. **25** (2000) 25–29
11. Boyle, E.I., Weng, S., Gollub, J., Jin, H., Botstein, D., Cherry, J.M., Sherlock, G.: GO::TermFinder—open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. Bioinformatics **20**(18) (2004) 3710–3715
12. Bok, J., Dolson, D.K., Hill, P., Ruther, U., Epstein, D.J., Wu, D.K.: Opposing gradients of Gli repressor and activators mediate Shh signaling along the dorsoventral axis of the inner ear. Development **134**(9) (2007) 1713–1722
13. Hunt-Newbury, R., Viveiros, R., Johnsen, R., Mah, A., Anastas, D., Fang, L., Halfnight, E., Lee, D., Lin, J., Lorch, A., McKay, S., Okada, H.M., Pan, J., Schulz, A.K., Tu, D., Wong, K., Zhao, Z., Alexeyenko, A., Burglin, T., Sonnhammer, E., Schnabel, R., Jones, S.J., Marra, M.A., Baillie, D.L., Moerman, D.G.: High-throughput in vivo analysis of gene expression in caenorhabditis elegans. PLoS Biology **5**(9) (2007) e237
14. Lein, E.S., Zhao, X., Gage, F.H.: Defining a Molecular Atlas of the Hippocampus Using DNA Microarrays and High-Throughput In Situ Hybridization. J. Neurosci. **24**(15) (2004) 3879–3889
15. Piper, J., Rutovitz, D.: Data structures for image processing in a C language and Unix environment. Pattern Recognition Letters **3** (1985) 119–129