



universität  
wien

# DIPLOMARBEIT

Titel der Diplomarbeit

**Establishing a Screen for Enhancers  
active in *Drosophila* Embryogenesis**

Verfasser

Gerald Stampfel

angestrebter akademischer Grad

Magister der Naturwissenschaften (Mag. rer. nat.)

Wien, März 2011	
Studienkennzahl	A490
Studienrichtung	Molekulare Biologie
Betreuer	Prof. Dr. Michael Jantsch



# Declaration of Authorship

I hereby declare that this thesis is my own work except when otherwise indicated. The methods described in Chapter 3 have been established together with *Evgeny Kvon* who planned and guided all the experiments. I hereby gratefully acknowledge his help and supervision.

Vienna, May 13, 2011

Gerald STAMPFEL

# Abstract

Enhancers are critical determinants of spatio-temporal gene expression. An enhancer's sole DNA sequence, cloned upstream of a reporter gene, is sufficient to drive expression which partially or fully resembles the endogenous gene's pattern. The sequence determinants which give rise to this specific activity are, however, unclear. While previous research concentrated on dissecting regulatory regions of individual genes, a large-scale approach might be needed to find shared sequences among enhancers with similar activity and thereby improve our understanding of the regulatory code.

This thesis reports on a method for the identification and analysis of enhancers active in the embryogenesis of the common fruit fly *Drosophila Melanogaster*. Upon screening 981 enhancer candidates using an *in-vivo* reporter assay, we find 403 (41%) to be active in at least one stage of development. Additionally, we find the fraction of active elements to increase as the developing embryo becomes more complex over time. For further analysis, we developed a computational pipeline enabling us to find elements of similar spatio-temporal activity and visualize their pattern over time and space. Overall, our results provide a reliable basis for future analysis which may lead to the identification of sequence elements determining an enhancer's specific activity.

# Kurzfassung

Enhancer sind massgeblich an der Regulation zeitlicher und räumlicher Genexpression beteiligt. Ein starker Indikator für die Relevanz von Enhancern ist die Beobachtung, dass die DNA-Sequenz eines Enhancers gekoppelt an ein Reporter-gen zu einem gewebe-spezifischen Expressionsmuster dieses Gens führt. Die essentiellen Teile einer Enhancer Sequenz die dessen Funktion zugrunde liegen sind jedoch noch nicht bekannt. Um dieses Problem genauer zu ergründen, ist die Analyse einer grossen Anzahl an systematisch verifizierten Enhancern nötig. Da sich die bisherige Forschung jedoch weitgehend mit regulatorischen Regionen einzelner Gene in unterschiedlichen experimentellen Versuchsanordnungen und Organismen beschäftigt hat, existiert eine derartige Ressource noch nicht.

Daher stellen wir eine Methode zur systematischen Identifizierung von embryonalen Enhancern in der Taufliege *Drosophila Melanogaster* vor. Insgesamt haben wir 981 potentielle Enhancer mithilfe von Reporter-Assays auf ihre zeit- und gewebsspezifische Aktivität hin untersucht. Eine unserer zentralen Beobachtungen zeigte einen graduellen Anstieg der Zahl aktiver Enhancer mit fortschreitender Embryonalentwicklung. Diese Beobachtung lässt sich durch Arbeiten anderer Forschungsgruppen erklären, wonach im Laufe der Embryogenese sowohl die Zahl der Gewebe als auch die Komplexität des Genexpressionsmuster in den einzelnen Geweben zunimmt. Der dadurch entstehende Bedarf an zusätzlicher Regulation erklärt den beobachteten Anstieg aktiver Enhancer in späten Embryogenese-Stadien. Um eine weitergehende Analyse unserer untersuchten aktiven Enhancer, welche unterschiedliche zeit- und gewebsspezifische Aktivitätsmuster zeigen, zu ermöglichen, wurde von uns ein computergestütztes Bildanalyseverfahren entwickelt. Dadurch konnten unsere Proben automatisiert auf zeitliche und räumliche Aktivität analysiert und visuell dargestellt werden. Zusammenfassend stellen unsere Resultate eine zuverlässige Ausgangsbasis für weitere Analysen dar die in der Zukunft zum genaueren Verständnis des zentralen Zusammenhangs zwischen der Sequenz und der Aktivität eines Enhancers beitragen können.

## Thanks to . . .

- . . . *my Family* for continuously supporting all the (weird) things i do in my life.
- . . . *my Friends* for making my life fun in general and in particular *Chriz* for being a confused genius, *Evil Hias* for saving this world from green energy, *Dr. K* (♀) for making me a friend of a wikipedia-known study and book author, *Dr. K* (♂) for giving me the opportunity to train myself in creating bar plots in excel and fruitful competition in life, *Marie* for being a critical mind and listening to my everyday life stories, *Norb* for important lessons on how to avoid being single (for a few hours at least), *Sarah* for repeatedly reminding us of what cultural and social anthropology actually is, *Surfinstructor* for well-researched fashion advises and *Thomaso Buscetti* for regular shelter in his flat and concurrently sharing his enormous gin collection.
- . . . *my Colleagues in the Stark Lab*. *Antonio* for sharing my perception of proper time management, *Anais* for knowing everything about R and continuously keeping all of us awake by being really nervous about dirty coffee cups, *Christian* for general awesomeness (1), long hair (2), and his car (3), *Daniel* for knowing everything about R (“of course”) and providing deeper insights into how it is like to sit next to an Autist, *Omar* for being a rather polite guy and knowing the number of SNPs in the human genome by heart, *Robert* for providing interesting aspects on how society can still profit from people who already passed away, *Evgeny* and *Dasha* for regular lessons on the stunning truth about Russian culture, *Kathi* for pleasant Friday-morning breakfasts on a regular basis, *Cosmas* (also for cigarettes; red Gauloises, soft pack) and *Christoph* for cracking the regulatory code of S2 cells, and *Michi* and *Martina* for being the rational and well-caring moms of this crazy bunch of people.
- . . . *the Head of the Lab, Dr. Alexander Stark*, for his contagious passion for science and solving problems fast and simple.
- . . . *my Study Coordinator Dr. Hamilton* for spending a considerable amount of neurons in pushing me through my undergraduate studies.

# Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. Related Studies and Resources</b>	<b>3</b>
2.1. ENCODE and modEncode Projects . . . . .	3
2.2. REDFly . . . . .	3
2.3. BDGP <i>In-Situ</i> Database . . . . .	3
2.4. Enhancer Screens . . . . .	4
2.5. Enhancer Library “Vienna Tiles” . . . . .	4
<b>3. Screen Setup</b>	<b>7</b>
3.1. Expansion of Transgenic Flies . . . . .	7
3.2. Embryo Collection and Tissue Fixation . . . . .	8
3.3. Staining . . . . .	8
3.4. Image Acquisition . . . . .	12
<b>4. Computational Methods for Image Analysis</b>	<b>15</b>
4.1. Image Processing with MATLAB . . . . .	15
4.2. Image Segmentation . . . . .	17
4.3. Embryo Orientation . . . . .	21
4.4. Embryo Registration . . . . .	24
4.5. Pattern Extraction . . . . .	26
4.6. Pattern Comparison . . . . .	27
4.7. Clustering of Images with Similar Patterns . . . . .	32
<b>5. Results on Enhancer Activity</b>	<b>39</b>
5.1. Positive Rate . . . . .	39
5.2. Spatio-Temporal Activity . . . . .	40
5.3. Genes - The BDGP Dataset . . . . .	43
5.4. Spatio-Temporal Additivity . . . . .	47
<b>6. Discussion</b>	<b>51</b>
<b>A. Materials and Methods</b>	<b>53</b>
A.1. Solutions and Commercial Reagents . . . . .	53
A.2. <i>In-Situ</i> Probe Generation . . . . .	53
<b>B. Curriculum Vitae</b>	<b>57</b>
<b>Bibliography</b>	<b>59</b>





# 1. Introduction

Genes provide the blueprints for making proteins which are the chemical workhorses inside every cell. Proteins form a cell's structure and perform most of its functions such as breaking down toxic substances in the liver or storing oxygen in red blood cells. In order to produce a certain type of protein, a gene's DNA sequence is first transcribed into RNA. In a process called translation, RNA then directly serves as an instruction set for protein assembly. Transcription and subsequent translation of a gene are together referred to as *gene expression*.

Even though almost all cells in our body contain the same DNA, different types of cells produce different proteins in a strictly controlled process [62, 61]. Examples of differentially expressed genes include the alcohol dehydrogenase in hepatocytes of the liver and hemoglobine present in erythrocytes of the blood. If this regulation fails, a cell can no longer fulfill its tasks and disease is the consequence [33]. Diseases linked to misregulation of genes include autism [9] and several types of thalassemia [65, 32, 18]. Additionally, gene regulation is especially important during development where cells have to act *at the right time and place* in order to together assemble a whole organism [17, 35].

Gene expression is controlled at the levels of transcription and translation. Especially the *start of transcription* is an important regulatory step. Specialized regulatory elements, usually located in the vicinity of a gene, play an important role in this process. These elements, so-called *enhancers* [6], are responsible for activating transcription at a certain time and place. Thereby, a gene's enhancers contribute to its overall spatio-temporal expression pattern in a modular fashion [56, 55, 66].

Apart from being *modular*, enhancers have been demonstrated to work *context independently*. This has been shown in reporter gene experiments where a fragment of DNA contained all the regulatory information necessary [3]. The language of DNA is based on a simple four-letter alphabet: **A**, **T**, **C**, and **G**. It has been shown that protein-coding genes are made up of defined three-letter words (e.g. **GCA**), so called codons [15]. Codons specify the chemical composition of a protein. Additionally, the boundaries of a protein-coding gene are marked by specific start- and stop-codons. This so-called *genetic code* is therefore well-defined and enables us to locate genes in a given DNA sequence and even predict the encoded proteins. Enhancers, in contrast, have been found to be composed of differently sized words, so called motifs. These Motifs serve as binding platforms for a specialized class of proteins called transcription factors (TFs) which control tissue-specific transcription [16]. Unfortunately, motifs are not arranged according to fixed rules on the DNA, they might be separated, adjacent, or even overlapping. Furthermore, there are no motifs known marking the start or end of an enhancer.

In order to decipher this seemingly more complex *regulatory code*, a large set of active enhancers might be needed. However, the activity of an enhancer is not binary but specific for a certain time and place. An ideal resource would therefore comprise a set of enhancers and their spatio-temporal activity. The goal of this thesis was to establish a method for creating such a novel resource.

This thesis introduces an approach for the large-scale identification of developmental enhancers in the common fruitfly *Drosophila Melanogaster*. We chose *Drosophila* because it is a well-established model organism allowing us to study the conserved process of transcriptional regulation. We established an *in-vivo* reporter assay for testing intergenic and intragenic fragments of the non-coding genome for their regulatory activity. Each reporter construct contains a DNA fragment cloned upstream of a minimal promoter and a reporter gene. We used an existing library of transgenic flies, each carrying one reporter construct in a defined genomic location. In fly embryos, we recorded reporter gene transcription and thereby obtained a spatio-temporal readout of enhancer activity.

The structure of this thesis is as follows: **Chapter 1** summarizes the motivation for this work, **Chapter 2** gives an overview of related studies and how our approach is different, **Chapter 3** explains the experimental procedures in detail. **Chapter 4** describes the methodological results, the development and validation of a computational pipeline for segmenting *in-situ* images and **Chapter 5** provides the biological results. **Chapter 6** summarizes this thesis and provides a future outlook.

## 2. Related Studies and Resources

In this chapter, we provide an overview of the previous and current work conducted on identifying cis-regulatory modules and the setup of our approach.

### 2.1. ENCODE and modEncode Projects

The Encode (Encyclopedia Of DNA Elements) project has the goal of mapping all the functional elements in the human genome. Encode was started in 2003 by the National Human Genome Research Institute (NHGRI) and published its first findings in 2007 [14]. modEncode was announced to be funded by the NIH in 2007 and is dedicated to the identification and validation functional elements in the model organisms *Drosophila melanogaster* and *Caenorhabditis elegans*. The main idea for starting modEncode was to also *in vivo* validate the functionality of candidate regions identified in large-scale experiments which is very difficult or not possible at all for humans. The first findings of the modEncode project have been published in late 2010 [23, 40].

### 2.2. REDFly

The REDFly database is a collection of Cis-Regulatory Modules (CRMs) and Transcription Factor Binding Sites (TFBS). This resource contains data collected from the literature by collaborating groups from the Universities of Buffalo and Manchester. It was initially released containing  $\approx 600$  CRMs in 2006 [21]. Its current version 3, released in 2010, comprises a collection of around 800 CRMs [20].

### 2.3. BDGP In-Situ Database

The Berkeley Drosophila Genome Project (BDGP) developed a high-throughput pipeline for *in-situ* staining of coding transcripts in *Drosophila*. Their aim is to stain, image, and manually annotate all the expression patterns of genes active in embryonic development. So far, their dataset holds 97 842 images for 7 153 genes [64].

This dataset is complementary to the data we are generating in this study and enables us, for example, to compare the activity of found regulatory elements with the expression pattern of adjacent genes. We therefore applied the computational means for analyzing *in-situ* images, developed in the course of this study, to the BDGP pictures (see Section 5.3).

## 2.4. Enhancer Screens

Previous approaches for the systematic identification of enhancers in a genome can be roughly classified into (1) *computational predictions* based on sequence conservation and motifs and (2) methods based on information about the *chromatin* landscape. Additionally, randomly sampling a genome for regulatory elements has been done too [12].

For *computational predictions*, the conservation of the DNA sequence between related species can be used in order to infer regulatory activity. This has been done for a variety of species such as *Drosophila* [59], humans [68, 49, 52] and other vertebrates [69]. However, not all enhancers are conserved [8, 28].

Since many transcription factors are known to bind to specific DNA sequences [54, 5, 71], another approach is to search the genome for clusters of motifs [2, 37, 47, 7].

Also the *chromatin* landscape is of value and can be assayed by performing Chromatin-Immunoprecipitations (ChIP) of certain proteins followed by microarray analysis (ChIP-Chip) or deep sequencing (ChIP-Seq). Binding sites or regions of chromatin components such as transcription factors, modified histones, or transcriptional coactivators are used to identify sites of regulatory potential [67, 53, 72, 27, 70].

Proteins are thought to bind to “open” chromatin. Another approach is therefore to identify these easily accessible regions by methods such as nuclease hypersensitivity [26, 38, 10, 48] or Formaldehyde-Assisted Isolation of Regulatory Elements (FAIRE) [43, 22, 24].

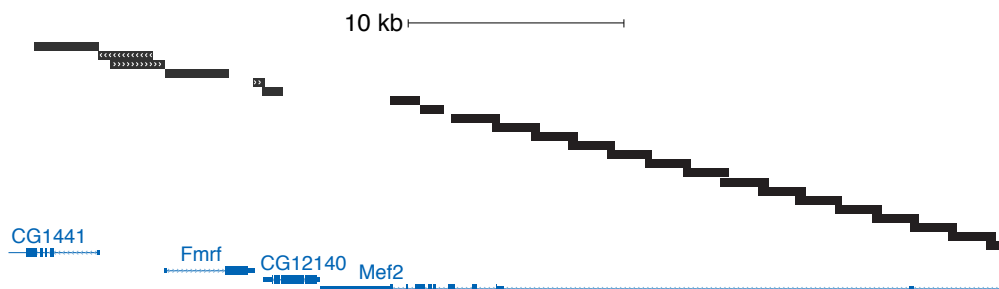
Enhancers are thought to physically interact with their target promoters by loops in the DNA [42, 50]. The three-dimensional conformation of DNA, identified by methods such as Chromosome Conformation Capture (3C) and its variants 4C and 5C, therefore also hints to the location of enhancers [39, 13, 25, 46, 57].

## 2.5. Enhancer Library “Vienna Tiles”

In the screen described in this document, we are probing the *Drosophila* genome using a unique resource created for the identification of regulatory elements.

We use a library which was designed for neurobiological purposes by Barry Dickson and Alex Stark at the Institute of Molecular Pathology (IMP) in Vienna, Austria. In order to gain genetic access to preferably small sets of neurons, an algorithm was devised for finding transcriptional enhancers active in the adult fly brain. On the computer, this library was constructed out of the *Drosophila* genome by excluding all coding and repetitive sequences and splitting the remaining non-coding genome into overlapping fragments of approximately 2kb size (see Figure 2.1).

With this strategy, the non-coding and non-repetitive parts of the *Drosophila* genome have been divided into 65 910 pieces or “tiles” with an average length of 1819bp and an average overlap of 450bp between adjacent tiles. These fragments are then amplified by PCR from genomic DNA and cloned into a



**Figure 2.1.:** VT Library: Example region of *Drosophila* genome tiling (modified from UCSC Genome Browser [31])

vector upstream of a *Drosophila* Synthetic Core Promoter (DSCP) [51] and *Gal4* which acts as a reporter gene. *Gal4* is a transcription factor active in galactose metabolism in yeast and widely used in *Drosophila* genetics in conjunction with its cognate DNA binding motif, the Upstream Activating Sequence (UAS) [11].

The whole reporter construct, including the putative enhancer fragment, DSCP, and *Gal4*, is then injected into germ lines of *Drosophila* embryos and thereby recombines into a specific locus in the genome. We can therefore check for the regulatory activity of a fragment in this standard chromatinized context by performing *in-situ* hybridizations to the *Gal4* transcript.

See Figure 2.1 for the tiling of an example locus, illustrating the strategy used. Shown is a region on chromosome 2R containing parts of the gene *Mef2* which is an important regulator of myogenesis. Also shown are the promoter regions of *CG1441* and *Fmr1* which are transcribed in opposite directions. In the tiling algorithm used, promoters are defined as the sequence 2kb upstream of the transcription start site (TSS). Since some promoters are thought to be directional, the DNA upstream of *CG1441* and *Fmr1* are present separately and direction-specific in the library (black bars marked with additional arrows in Figure 2.1). The non-promoter tiles shown in Figure 2.1 are cloned regardless of the direction (black bars).

Of all the fragments designed *in-silico*, only a subset has been cloned and injected into flies. Due to the initial purpose of this library, the fragments actually realized contain a bias towards genes expressed in the nervous system.

The experimental methods for screening this library are introduced in detail in Chapter 3.



## 3. Screen Setup

This chapter describes the experimental procedures in detail. This ranges from handling the transgenic flies to imaging of the stained embryos, including all the protocols and reagents used.

Experimental parts of the text introducing the detailed steps of a protocol are marked with a black bar on the left side of the text as in the following example:

### Experimental steps

- Step 1
- Step 2
- ..

### 3.1. Expansion of Transgenic Flies

In order to provide sufficient quantities of embryos, we have to propagate the transgenic fly lines which takes about two weeks. This is done by putting  $\approx 10$  flies in a Nipagin-containing bottle for two days and moving the flies on to a new bottle after two days. After another two days, the flies are moved once again so that we end up with three bottles. After additional ten days, new flies in all three bottles will start to hatch. The newly eclosed flies from these three bottles together will then be enough to populate an embryo collection cage.

Our collection cages are equipped with an opening at the bottom which is used to plug-in applejuice-agar plates (see Figure 3.1). These applejuice plates are covered with moist yeast pellets and serve as food for the flies which lay their eggs into the agar.

After a cage has been populated with flies, we leave the flies in the cage for two days before we collect the eggs on the third day. The applejuice plates are changed each day in the morning in order to provide fresh food. This procedure is a common practice and serves the purpose of letting the flies adjust to this new environment and thereby makes them lay more eggs during the actual collection process.

Before we start an overnight collection, we change the applejuice plates and let the flies lay eggs for two hours which is called “prelaying”. This is done in order to get rid of older eggs which have been held back by the females. We then switch to a new applejuice plate again and collect the resulting eggs after 13 hours which is usually done overnight in a 25° incubator. We chose 13 hours because this is the time needed for a *Drosophila* embryo to reach stage 16 of development [29] which is the latest stage we are interested in.

## 3.2. Embryo Collection and Tissue Fixation

As a prerequisite for the *in-situ* procedure, the *Drosophila* embryos have to be tissue-fixed. This is done by incubation with formaldehyde and described in detail in this section.

First, we need to collect the embryos from the applejuice plates. This is done by pouring  $\approx 1$ ml of distilled water into the plate, deattaching the embryos with a paintbrush, and pouring this embryo-water suspension through a sieve. We are using custom-made 12 well plates with embedded sieves in order to minimize the number of handled vessels (see Figure 3.1).

The plates are then incubated in 50% bleach for three minutes in order to remove the chorion of the eggs. Thorough washing with distilled water is needed to get rid of the remaining bleach. Constant agitation of the embryos during dechoriation is needed in order to prevent the formation of clumps.

The embryos are then transferred from the sieves into scintillation vials (see Figure 3.1) containing 7.5ml Heptane and 7.5ml 4% Formaldehyde (see Section A.1 for preparation). In order to fixate the tissues, the scintillation vials are put on a shaker at maximum speed for 20 minutes.

After the fixation, the formaldehyde (bottom phase) is removed from the scintillation vials with a glass pipette. Try not to take embryos when removing fixative (they should settle at interphase). Then, 7.5ml Methanol are added and the vial is manually shaken very hard for 20 seconds in order to remove the embryos' vitelline membrane. All the non-damaged embryos now slowly move to the ground of the vial whereas all damaged ones stay in the upper phase (Heptane).

The embryos are now fixed and transferred to autoclaved Eppendorf tubes for later staining. This is done by carefully removing the upper phase (Heptane) of the vial and major parts of the bottom phase (Methanol), including all the damaged embryos contained in the upper phase. The remaining embryos on the bottom of the vial can now be easily transferred to 1.5ml tubes. After washing for three times with Methanol, the embryos can be stored for up to one year at  $-20^{\circ}\text{C}$ .

## 3.3. Staining

The staining process which includes post-fixation, protease treatment, probe hybridization, and alkaline phosphatase (AP) staining is done in 96-well plates. This procedure is described in detail in this section.

Most of this protocol is being conducted by a liquid handling robot (Bravo Automated Liquid Handling Platform, Agilent Technologies <sup>1</sup>).

See Appendix A.2 for details on how we generate the *in-situ* probe.

### Proteinase K treatment and postfixing

- aliquot  $\approx 50\mu\text{l}$  of embryos into  $200\mu\text{l}$  tubes or 96 well plate

<sup>1</sup><http://www.home.agilent.com>



- rinse embryos with
  1. Methanol
  2. Methanol:PBT 1:1
  3. 2 times with PBT
- rinse in 4% formaldehyde
- postfix for 20' in 4% formaldehyde, rotating
- wash 3x with PBT for 2' each
- rinse in 100 $\mu$ l of proteinase K solution (9 $\mu$ g/ml of proteinase K in PBT)
- add 100 $\mu$ l proteinase K solution to each sample <sup>2</sup>
- incubate for 13' at RT, mix 5-6 times during this period with pipet
- incubate for 1h on ice
- remove proteinase K solution
- wash 2x for 2' each with 2mg/ml glycine solution, rotating
- wash 2x with PBT
- rinse in 4% formaldehyde
- postfix again for 20' in 4% formaldehyde, rotating
- wash 5x with PBT for 2' each

### **Probe hybridization**

- rinse embryos with
  1. PBT:RNA hybridization solution 1:1
  2. RNA hybridization solution
- prepare prehybridization solution:
  - boil 100 $\mu$ l hybridization solution per sample for 5' at 100°C
  - cool on ice for at least 5'
- remove hybridization buffer
- rinse embryos in cooled prehybridization solution
- add cooled prehybridization solution to the samples
- 2h incubate for at least 2h at 56°C
- prepare probe solution:
  - 3ng/ $\mu$ l probe in hybridization solution

- heat for 3' at 80°C<sup>3</sup>
- cool for at least 5' on ice
- remove prehybridization solution
- rinse embryos in probe solution
- add probe solution and incubate o/n at 56°C

### **Washing**

- heat PBT and hybridization buffer to 56°C
- remove probe solution
- rinse embryos with 100µl prewarmed hybridization buffer
- add again 100µl prewarmed hybridization buffer
- incubate 2x for 30' at 56°C
- wash embryos, 20' each step, at 56°C
  1. hybridization buffer:PBT 3:1
  2. hybridization buffer:PBT 1:1
  3. hybridization buffer:PBT 1:3
- wash 4x for 5' each with prewarmed PBT
- cool embryos to RT

### **Alkaline phosphatase (AP) staining**

- rinse embryos in PBTB
- block by incubating embryos for 1h with PBTB, rotating
- rinse in anti-DIG antibody solution (1:2000 from Roche in PBTB)
- incubate o/n with anti-DIG antibody solution, rotating (1:2000 from Roche in PBTB)
- rinse embryos in PBTB
- wash 1x with PBTB for 30', rotating
- wash 2x with PBT for 30' each, rotating
- wash 2x with AP buffer for 10' each, rotating
- transfer to microscope-compatible wells (regular 24 well plates for example)



**Figure 3.1.:** Equipment used for collecting and fixing embryos.

- remove AP buffer
- add developing solution ( 1ml)
  - Roche BM Purple ready-to-use solution <sup>4</sup> or
  - Roche, 45  $\mu$ l NBT and 35  $\mu$ l BCIP per 10ml of AP Buffer or 180  $\mu$ l of NBT/BCIP Roche combined solution per 10ml AP buffer <sup>5</sup>
- incubate in the dark with gentle shaking until desired color is achieved <sup>6</sup>
- remove developing solution
- rinse 3x in PBT
- rinse 5x with 100% ethanol for 2 , 2 , 2 , 30 , and 2
- rinse 3x in PBT
- transfer to tubes/well plates using PBT
- add 70% glycerol in PBS
- ready. store at 4 C rinse embryos in PBTB

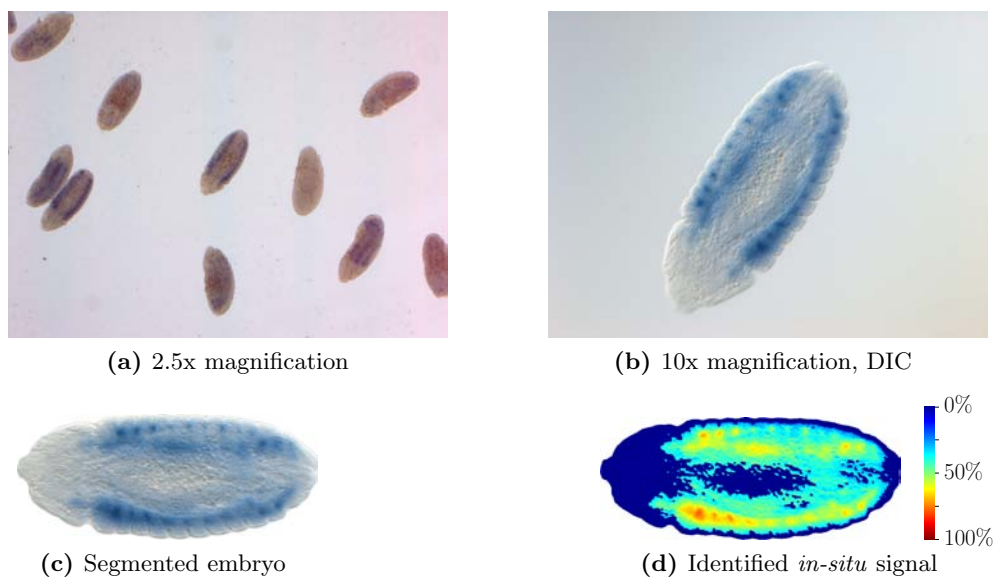
### 3.4. Image Acquisition

We then mount the embryos on a regular slide and document the *in-situ* signal by taking images of the embryos on a microscope. For each of the transgenic fly lines, i.e. each carrying the reporter for one specific region of the non-coding genome as explained in Section 2.5, we try to take at least three images for (1) each of the orientations lateral, dorsal, and ventral and (2) each of the developmental stage groups 3-6, 7-8, 9-10, 11-12, and 13-16. However, we only document a certain stage if our reporter shows activity in this stage.

Concerning our microscope setup, we are taking images using Differential Interference Contrast (DIC) which is a method for enhancing contrast in transparent samples. Image acquisition is done at 10x magnification with a digital camera connected to the microscope. To each image, we then manually assign the metadata necessary for subsequent processing: A unique identifier of the genetic line and the orientation and developmental stage of the documented embryo.

Figure 3.2 shows examples of how our input images look like and the results of the subsequent image processing. Figure 3.2 (b) shows how a typical raw microscopic image looks like, in this case it would be annotated as transgenic line VT33937, stage 13-16, and laterally oriented.

In order to extract the relevant information out of each of these images, we devised computational means to automatically segment *Drosophila* embryos and analyze the *in-situ* signal as shown in Figures 3.2 (c) and (d). This pipeline is explained in detail in Chapter 4.



**Figure 3.2.:** Examples of microscopic images at different magnifications and the corresponding results of embryo segmentation and pattern extraction (see Chapter 4). Each pixel in the heatmap in (d) is stained according to the *in-situ* signal intensity in this region, ranging from 0% (dark blue) to 100% (dark red).



## 4. Computational Methods for Image Analysis

In order to conduct meaningful comparisons between the various transgenic *Drosophila* lines, the relevant information in each microscope image (termed “raw image” in the remainder of this chapter) has to be extracted and the resulting embryos aligned in a common way in order to systematically find similarities among the expression patterns.

We first introduce a few basics of how image processing is done in MATLAB, then move on to explain how we segment the image in order to find and align the embryo, project it onto a common template embryo, extract the *in-situ* signal, and, finally, cluster the images by similarity of the expression pattern of our reporter gene *Gal4*.

### 4.1. Image Processing with MATLAB

In this section we shortly introduce a few basic image processing methods and properties of the MATLAB image processing toolbox in a non-exhaustive manner.

#### 4.1.1. Color- and Grayscale Images

Images are represented in MATLAB as multidimensional matrices. A black and white picture of 10x10 pixels for example is represented by a [10 10] matrix where each data point is either 0 or 1. Grayscale- and color images on the other hand allow discrete values between 0 and 255 (or continuous between 0 and 1). An RGB picture of 10x10 pixels is represented by a [10 10 3] matrix where each of the three third dimension matrices represents one of the channels red, green, and blue.

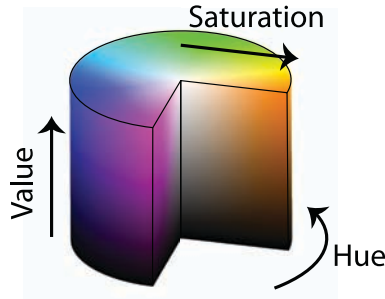
#### Hue Saturation Value (HSV) Colorspace

In an RGB image, the color of each pixel is defined by three values: red, green, and blue. In an HSV image, the three components are: hue, saturation, and value (see Figure 4.1):

**Hue** Originally described as “the degree to which a stimulus can be described as similar to or different from stimuli that are described as red, green, blue, and yellow” [19]. Non-technically, one might describe it as the color itself, such as if it is red or green etc. Range: Discrete  $0^\circ - 360^\circ$ .

**Saturation** The colorfulness of a stimulus relative to its own brightness [19].  
Range: Continuous 0 – 1.

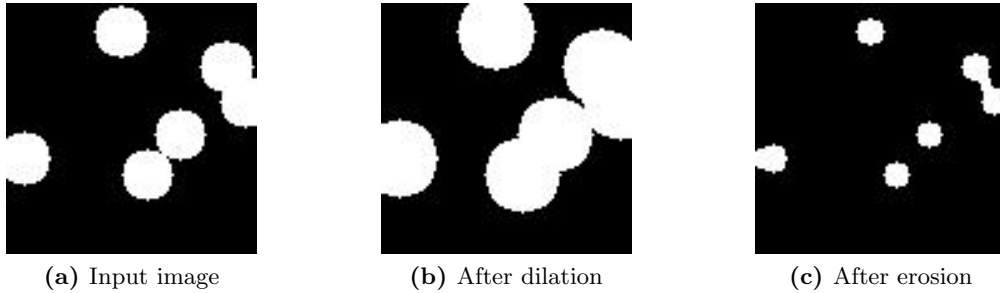
**Value** Reflects how bright a color is. Range: Continuous 0 – 1.



**Figure 4.1.:** The Hue Saturation Value (HSV) colorspace is used during the extraction of the *in-situ* signal explained in Section 4.5.

#### 4.1.2. Erosion and Dilation

Erosion and dilation are two image operations commonly used during image segmentation and other processes described in this chapter. As can be anticipated from the name, image dilation extends an area, whereas erosion contracts it. An example for a binary image is shown in Figure 4.2. In this example, circles which are very close to each other or touching are connected by dilation and disconnected by erosion.

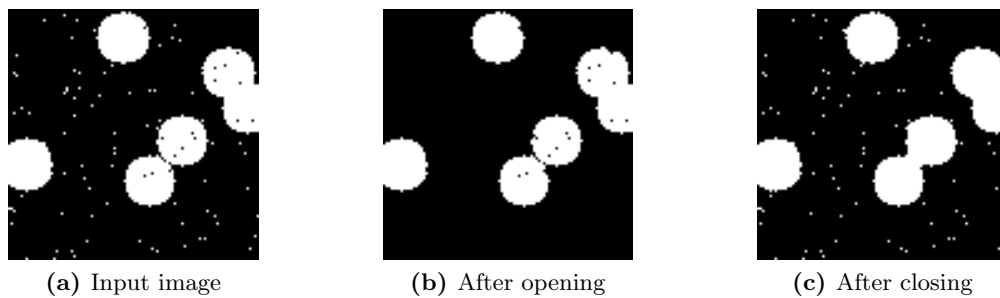


**Figure 4.2.:** Dilation and erosion are two basic methods forming the basis of higher order operations explained in Section 4.1.3.

#### 4.1.3. Morphological Opening and Closing

Opening and closing are successive erosions and dilations. Opening is an erosion followed by a dilation and closing is dilation-erosion. The result of a morphological opening is the conversion of small and isolated foreground objects to background (see Figure 4.3 (b)) and closing does the opposite for isolated background objects (see Figure 4.3 (c)).



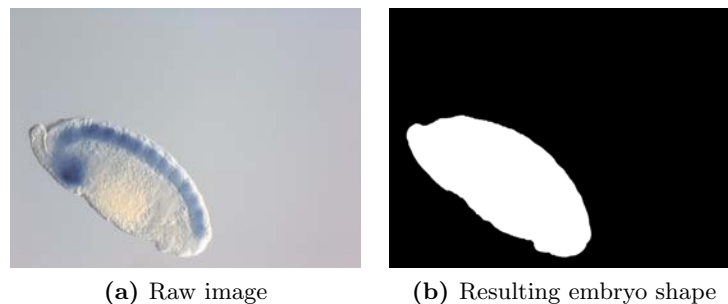


**Figure 4.3.:** Morphological opening and closing. These methods are used extensively in the processes described in this chapter.

Morphological opening and closing are used during image segmentation (see Section 4.2).

## 4.2. Image Segmentation

The goal of this step in our computational workflow is to identify the embryo of interest in the raw picture. We apply a few basic steps in order to identify and subtract the background, determine which of the found shapes on the picture are indeed complete embryos, and, if necessary, separate touching embryos. Finally, following the segmentation process, each identified embryo is rotated to horizontal. See Figure 4.4 for a short overview of this process which will be explained in detail in the following paragraphs.



**Figure 4.4.:** Overview of our segmentation process. (Note: See Section 4.3 for details on the process of orienting the embryo picture properly)

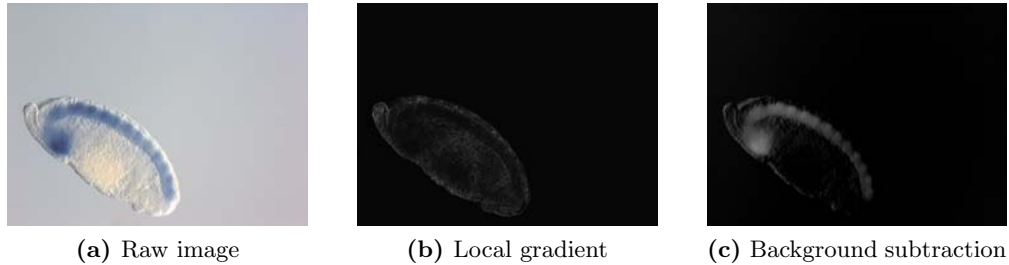
### 4.2.1. Foreground-Background Separation

We use two basic assumptions to separate fore- from background:

**Local Environment:** There is more change around each pixel in the foreground (i.e. inside the embryos) than in the background.

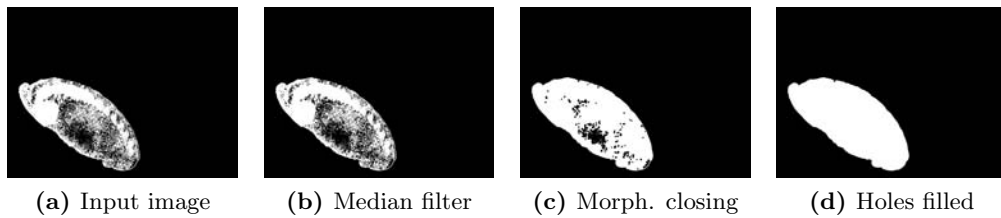
**Color:** Most of the pixels in an image belong to the background.

Therefore, we (1) calculate the *change around each pixel* in order to obtain a gradient grayscale image and (2) find the *median intensity for each of the three color channels* and subtract this value, which is the value of a pixel belonging to the slide background, from the original picture (see Figure 4.8).



**Figure 4.5.:** Foreground/background separation is done by analyzing local gradients and subtraction of the background.

We then obtain the binary embryo mask by separately thresholding each of the two calculations (change around each pixel and background subtraction) using Otsu’s method [45] and combining the results with a logical OR. This picture is then subjected to a median filter, morphological closing (explained in Section 4.1.3), and filling of all holes left in the shape (see Figure 4.6).



**Figure 4.6.:** Shape refinement conducted to properly identify whole embryos.

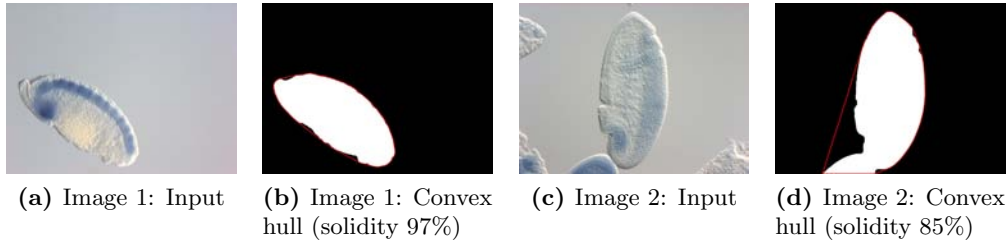
At this point, we discard all the small areas in the binary mask and continue working only with the five biggest identified shapes since in general we expect only one final embryo on each picture. If the embryo of interest on this picture is not touching and cleanly separated from other embryos which might be on the picture, all the necessary work is done for this picture. Therefore, the following steps are concerned with finding out if the shape identified is a single embryo or multiple embryos which have to be separated.

#### 4.2.2. Shape Selection

The criteria for determining if a candidate shape is accepted as a proper embryo or not are (1) the *size of the area*, (2) the *solidity*, and (3) if the shape is *touching the border*. See Figure 4.8 for an overview of the algorithm.

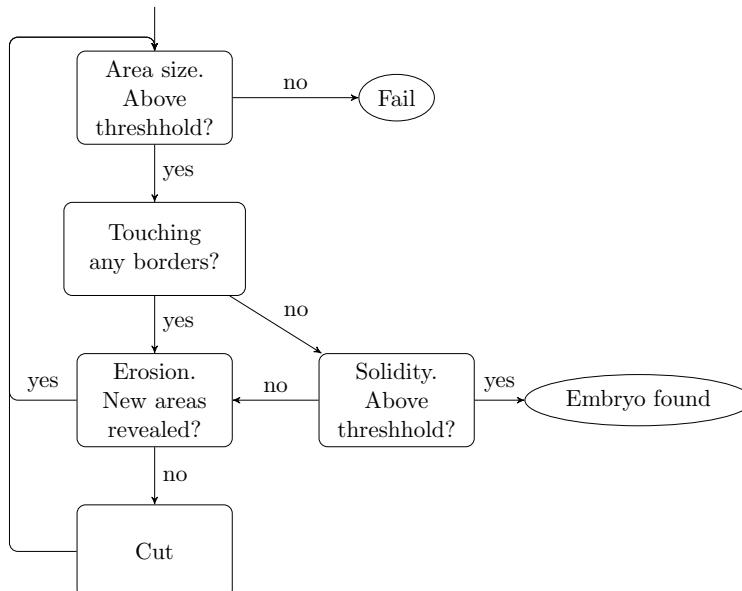
The solidity of a shape is defined as the area which is covered by its *convex hull*. The convex hull can be thought of a rubber band put around a shape, it will follow rounded outside contours tightly but connect indentations with a

straight line. See Figure 4.7 for two example images and their corresponding convex hulls.



**Figure 4.7.:** The *solidity* is defined as the area under the convex hull of a shape. We use this value to assess whether a candidate shape contains only one or multiple connected embryos.

As can be seen in Figure 4.8, our algorithm first analyzes the size of all candidate shapes identified in an image. If a certain area cutoff is not met, the shape is discarded and the next shape in the image is considered. Alternatively, if the area is big enough, it is intersected with the borders of the whole picture. Due to the way we take images on the microscope, the embryo of interest can not be directly on the border. All shapes intersecting the image border are then subjected to *erosion* and/or *cutting* (both explained in Section 4.2.3). If the area under inspection is not touching the border but very likely contains multiple embryos, as concluded from its solidity (see Figure 4.7), embryo-separation by erosion or cutting will take place as well.

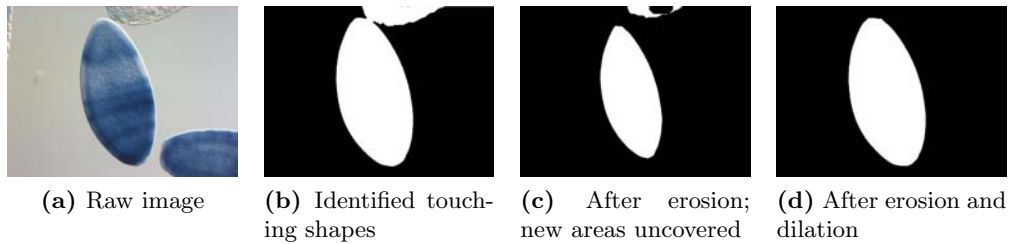


**Figure 4.8.:** Our object selection algorithm depicted as a flow chart.

### 4.2.3. Embryo Separation

In order to separate two embryos, we apply two different strategies: Erosion and cutting.

First, the shape, such as the one shown in Figure 4.9 (b), is *eroded* (explained in Section 4.1.2). If the erosion uncovers new areas, i.e. two embryos got separated, we dilate these two shapes separately again to recover the original, but this time, separated, shapes. See Figure 4.9 for an example of embryo separation by erosion.

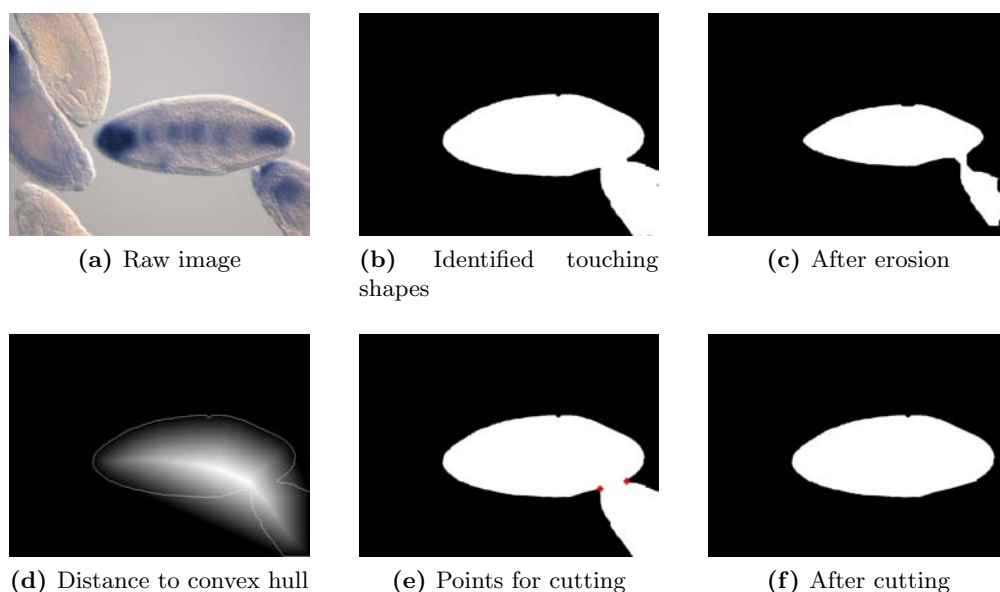


**Figure 4.9.:** Touching embryos are separated by *erosion*.

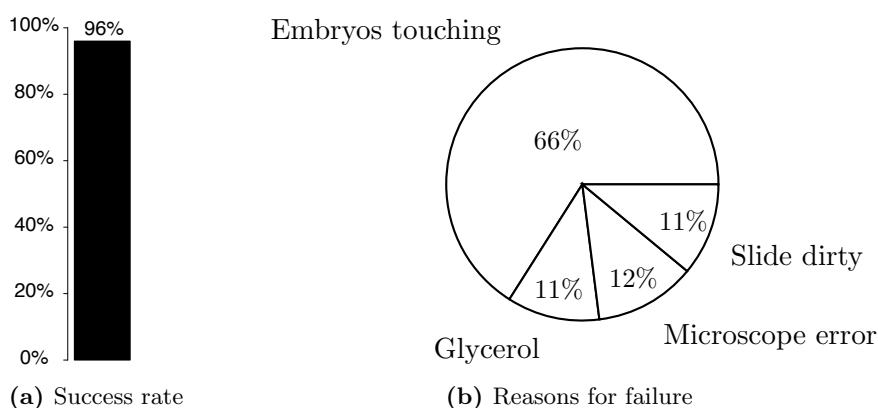
If erosion does not identify any new areas, we continue by trying to separate the putative embryos with a straight line. In this chapter we refer to this process as *cutting*. With cutting we make use of the convex hull again (see Figures 4.7 (b) and 4.7 (d)). See Figure 4.10 for an overview of the cutting process. In order to separate two embryos by a single cut, we need to find the two points on the shape outline that define the start and the end of the cut. By traversing all the points in the shape outline and calculating the distance to the convex hull, we are able to identify the constrictions in the shape as the points farthest from the convex hull (see Figure 4.10 (e)). By setting all the pixels between the two identified points to background, we obtain the shape of the isolated embryo (see Figure 4.10 (f)).

### 4.2.4. Performance

In order to quantify the performance of our image segmentation software described so far, we show a few numbers here on how well our DIC pictures are processed: For a manually curated test set of 3883 images, the software was able to find and properly crop an embryo in 94% (3648) of the raw images (see Figure 4.11 (a)). In 6% (235) of the cases, either no embryo was found or the embryo was aberrantly segmented (e.g. cut in the middle). In the majority of the failed cases, the separation of touching embryos did not work (66%, see Figure 4.11 (b)). Other reasons for failure are when the border of leaked glycerol touches the embryo (11%), microscope camera errors (12%), and dirty slides, aberrant focal planes, or a color gradient on the slide background (11%). See Figure 4.12 for example images out of each failure class.



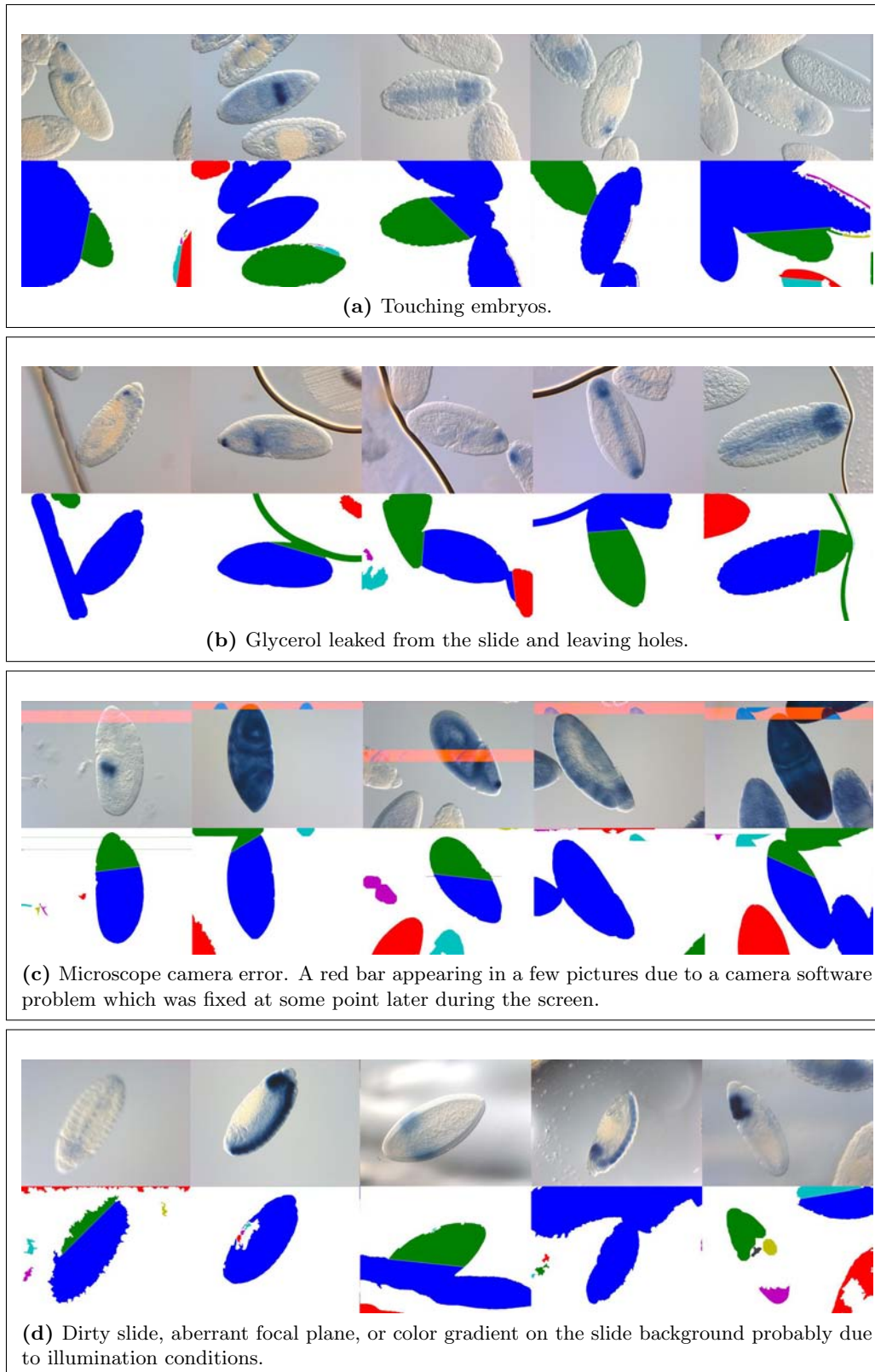
**Figure 4.10.:** Touching embryos are separated by *cutting*.



**Figure 4.11.:** Our segmentation algorithm performs well (94% success) on our test set as judged by manual inspection. For the remaining 6%, reasons for failure are touching embryos, microscope camera errors, glycerol leaking out of the slide or otherwise dirty slides.

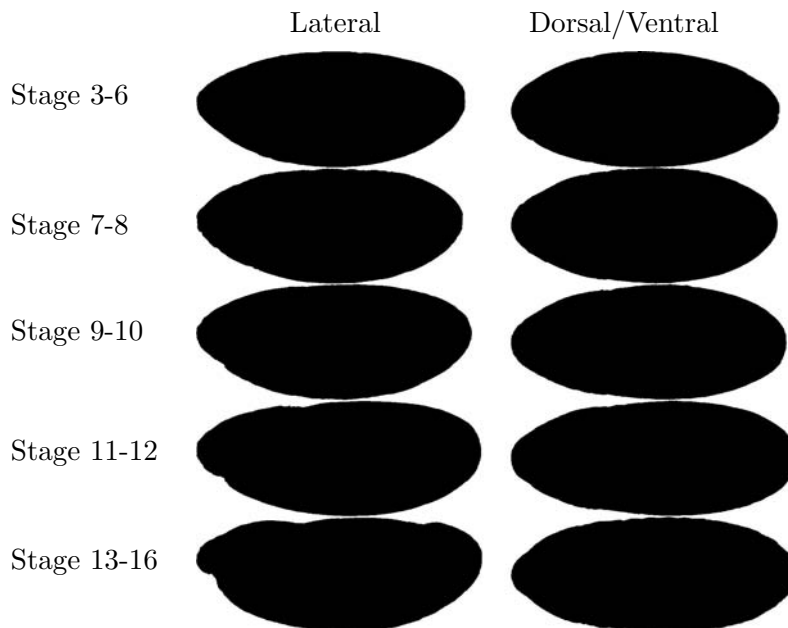
### 4.3. Embryo Orientation

After an embryo has been successfully identified in the raw image and rotated to horizontal, the embryo might be oriented with the anterior end either to the left or to the right. The same applies for the dorsal/ventral side which might be facing up or down. In order to align all embryos to a standard orientation, here anterior side left and dorsal side up, we set out to predict the current orientation of a given embryo shape by comparing it to a set of template shapes (see Figure 4.13). These stage-specific template masks have been created by



**Figure 4.12.:** Four major reasons for segmentation errors or missegmentation. Shown are pairs of pictures, the input picture (top) and the segmentation result of our software (bottom). The shapes identified by our algorithm are colored based on their size: Biggest area *blue*, 2<sup>nd</sup> biggest area *green*, 3<sup>rd</sup> biggest area *red*, 4<sup>th</sup> biggest area *turquoise*, etc.

averaging  $\approx 100$  individual, correctly-aligned, embryos.



**Figure 4.13.:** Master masks created out of  $\approx 100$  manually curated embryo images each used to predict the orientation of an input embryo shape.

An input embryo might be aligned in one of the following orientations:

- Anterior left, dorsal up (*desired orientation*)
- Anterior right, dorsal up
- Anterior left, dorsal down
- Anterior left, dorsal up

By combining an input mask with the corresponding template mask, rotated and flipped to recapitulate any of the four possible orientations just described, we can infer, using logical XOR, the current orientation of an input mask. As a result of the logical XOR operations we obtain four values quantifying how well the input mask fits to the template mask in (1) untouched-, (2) AP-flipped-, (3) DV-flipped-, and (4) AP-DV-flipped orientation. For a proper input embryo, we expect the value for the correct operation to be significantly lower than the others. Therefore, it is possible to quantify and calculate a *score* describing how reliable an orientation prediction was for a specific embryo picture:

$$score = \frac{minValue}{mean(otherValues)} \quad (4.1)$$

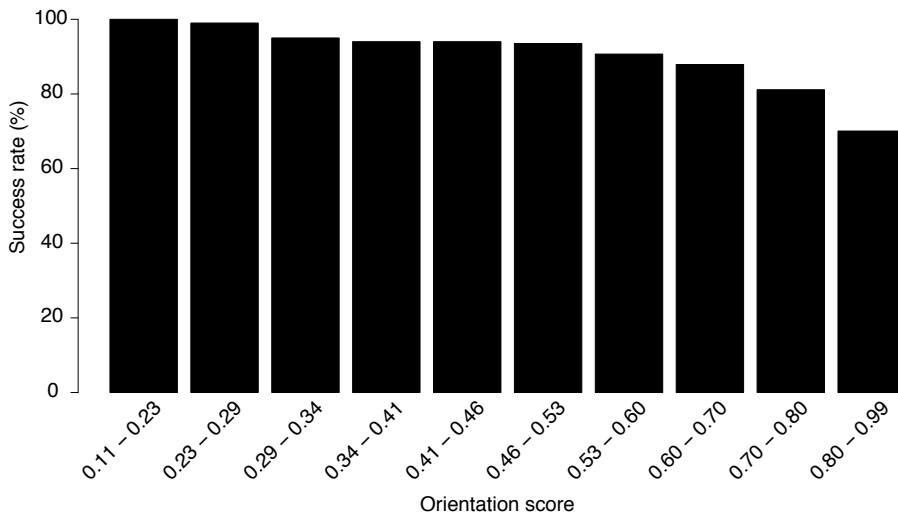
As shown in Section 4.3.1, this orientation score is a good predictor for the success of the orientation operation for the images generated in the course of this study.

### 4.3.1. Performance

In order to quantify the performance of the embryo orientation algorithm described above, we applied it to our set of manually curated correctly oriented embryo pictures. As a result, the automated orientation process was successful in 95% of the 3 548 pictures.

As can be seen in Figure 4.13, the success rate is higher for the late stage embryos due to the unique morphology at that time points: Stages 3-6 (76%), stages 7-8 (80%), stages 9-10 (91%), stages 11-12 (97%), stages 13-16 (96%), and all stages (95%).

The relationship between orientation score and success rate over all stages is shown in Figure 4.14. As expected, the success rate drops with poor orientation scores.



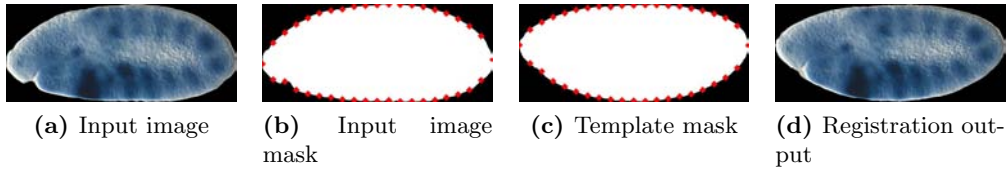
**Figure 4.14.:** Orientation score vs. success rate for all stages among ten equally sized bins of orientation score ranges.

## 4.4. Embryo Registration

The term registration in the context of image processing refers to the process of bringing different pictures into a common coordinate system. In the specific case of this study, we are aligning images of embryos onto a template embryo in order to be able to compare different patterns in the same embryo. This is necessary due to some heterogeneity of the embryos caused by the staining or mounting process. See Figure 4.15 for an overview of this process. The input embryo is shown in Figure 4.15 (a), embryo mask shown in Figure 4.15 (b), is registered onto the template mask shown in Figure 4.15 (c). The corresponding output of the registration process is shown in Figure 4.15 (d). As will be described in more detail in this section, the landmarks which are shown as red dots in Figure 4.15 are used to map the input embryo onto the template mask.

In order to perform registration, a template mask is needed onto which all





**Figure 4.15.:** Registration is used to compensate for small bumps or other heterogeneity of the embryo caused by the mounting or staining process. Shown in red are the landmarks chosen along the border of the embryo used to map coordinates in the input to the template image.

the input masks are registered. The embryonic development of *Drosophila melanogaster* is divided into distinct stages, whereas the morphology of the embryo differs from stage to stage. We therefore created the stage-specific template masks shown in Figure 4.13 by averaging approximately 100 individual, correctly-aligned, embryos.

As a first step, the landmarks on the embryo border of the input- and the template-embryo are chosen. We select (1) the most anterior and the most posterior points of the embryo and (2) 38 more points in between them, half of them on the dorsal side of the border and the other half on the ventral side (see Figure 4.15 (b) for example). In the same way, we choose control points on the template mask. We now have pairs of control points, each point in the input mask can be mapped to a point in the template mask.

Next, we save the information on how far the input and template control points are apart. More precisely, for each control point in the template mask, we save a vector to the corresponding control point in the input mask:

$$\vec{V}_{template} = CP_{input} - CP_{template} = \begin{pmatrix} x_{input} \\ y_{input} \end{pmatrix} - \begin{pmatrix} x_{template} \\ y_{template} \end{pmatrix} \quad (4.2)$$

The idea behind this is to map every pixel in the template mask to a pixel in the input mask which is known as *backward mapping*. At the current point in the process we have achieved this mapping for the control points only. We continue by interpolating all the points between control points on the embryo border. The interpolation is done by calculating the weighted average of the two closest control points. After this is done, we have a mapping for every coordinate of the border on the template mask to a coordinate on the input mask or picture.

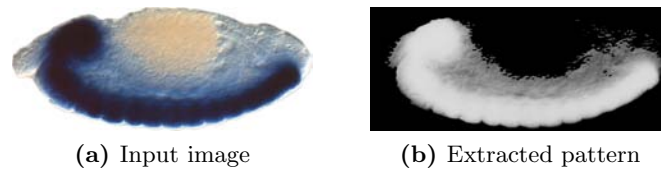
As a final step, we need to interpolate the inside of the template mask. This is done with the MATLAB function `roifill` which is part of the image processing toolkit. The general purpose of `roifill` is to fill in a region of interest in a grayscale image by “smoothly” interpolating inwards from the borders. In simple words, the author of the function describes it by saying “*imagine you that bend a loop of wire into a particular shape, dip the loop into a soapy solution, and then see how the resulting soap film smoothly fills in the region inside the loop*” [60]. In more technical terms, a region of interest is

filled by interpolating from the borders inwards using Laplace’s equation. After applying `roifill`, we are left with a fully filled template mask containing the vector to the coordinates of the corresponding input pixels. These vectors are then read out and the template mask is being filled one by one with pixels from the input image using bilinear interpolation.

The further processing steps in the computational pipeline such as the *in-situ* pattern extraction introduced in Section 4.5 are based on the registered images.

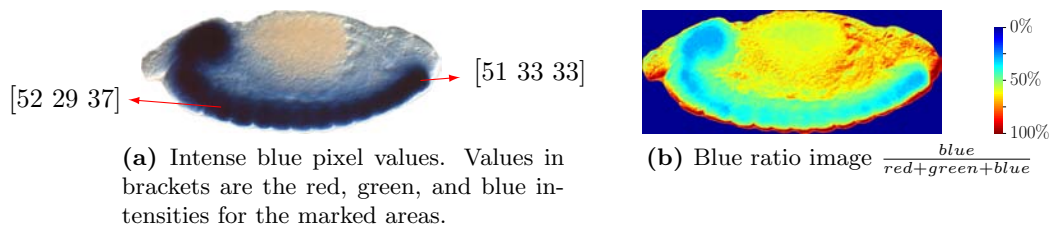
## 4.5. Pattern Extraction

After an embryo has been successfully detected, cropped, oriented, and registered onto a template embryo, the *in-situ* signal (i.e. the “pattern”) has to be extracted. See Figure 4.16 for an overview of this process.



**Figure 4.16.:** Overview of our pattern extraction process. The final result is a grayscale image of the pattern intensity.

The intuitive way to extract the signal would be to extract just the blue color or the blue ratio  $\frac{blue}{red+green+blue}$  of an input image. Unfortunately though, very intense blue staining can appear black and be represented as a dark form of just any color (see Figure 4.17 (a) for single pixel values [RED GREEN BLUE] of very intense staining). Therefore, when looking at a false color image highlighting the blue ratio, shown in Figure 4.17 (b) as a heatmap with the color code ranging from blue (low ratio) to red (high ratio), one can see that the regions of strong staining are actually depleted of blue pixels.

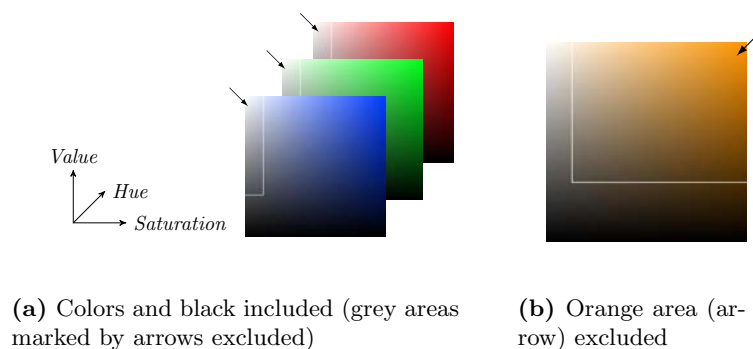


**Figure 4.17.:** Areas of intense staining show low blue ratio.

In order to overcome this problem of the intensely stained areas, we exclude all the colors which do not represent staining signal. We therefore filter out saturated gray and orange but at the same time include dark tones of any color. As a first step we convert the images from the Red Green Blue (RGB) colorspace to the Hue Saturation Value (HSV, introduced in Section 4.1.1)

colorspace. In the HSV colorspace, we select the colors of interest which can be seen as slicing out pieces of a cube with the three dimensions: hue, saturation, and value (see Figure 4.18).

As shown in Figure 4.18 (a), we first extract everything except the gray tones (marked by arrows in Figure 4.18 (a)) by setting thresholds for *saturation* and *value*. Next, the occasionally occurring orange background (see Figure 4.16 (a) for example) is excluded as shown in Figure 4.18 (b). As a final step, a noise filter is applied to exclude small disconnected areas of background staining.



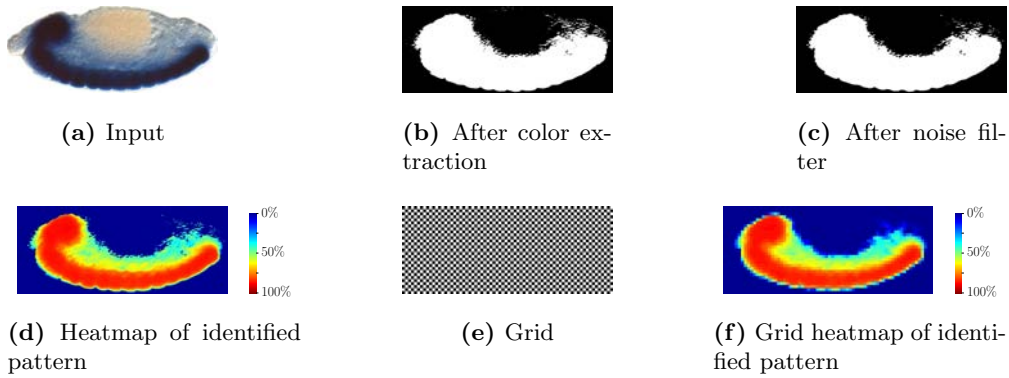
**Figure 4.18.:** The pattern extraction process works by excluding non-pattern background of gray and orange color.

Additionally, for all the further processing steps such as comparison we are not working with the actual pattern image just identified, but with a *grid representation of the pattern*. For this purpose we overlay the pattern image with grid cells and average all the pixel values beneath each grid cell. The purpose of this process, which is technically a resizing of the image, is to allow for some movement of a pattern which is given even in biological replicates due to variation in development, the cropping process, etc. All the following steps, described in the next section, are then working with this grid representation of the pattern.

See Figure 4.19 for the process of pattern extraction on an actual embryo picture.

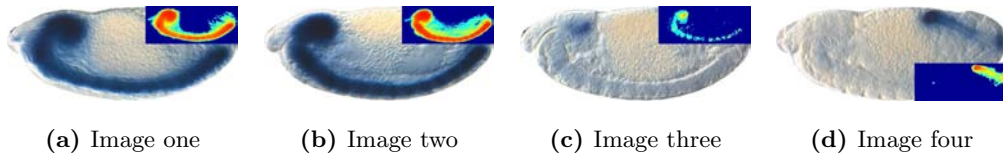
## 4.6. Pattern Comparison

All the steps explained so far such as finding and orienting the embryo, and extracting the *in-situ* pattern prepare the sample images for the actual comparison introduced in this section. The output of the comparison of two or more embryo images, such as the ones shown in Figure 4.20, is a quantification of how similar their patterns are. We always conduct comparisons of image pairs, it is therefore pairwise similarities what we are working with. For the images shown in Figure 4.20, we would expect a high similarity for the comparison (a) vs. (b), a lower value for (a) vs. (c) and the lowest value for (a) vs. (d). For this purpose, we devised a measure of similarity between two images which is based



**Figure 4.19.:** Pattern extraction process applied to an example image.

on the overlap of the corresponding staining patterns and therefore resembles an intuitive visual assessment of similarity.



**Figure 4.20.:** Pattern comparison. Example of three control comparisons yielding high similarity (a) vs. (b), medium similarity (a) vs. (c), low or no similarity (a) vs. (d).

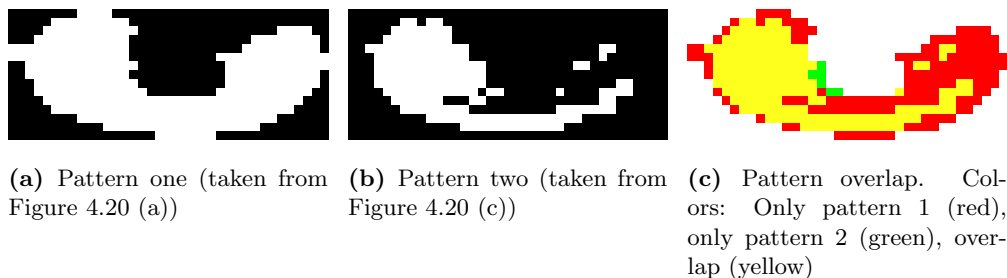
#### 4.6.1. Mutual Enrichment

The principle of how we determine the similarity between two images is the *mutual enrichment of their patterns*. When comparing two patterns, which for simplicity we assume to be binary, pattern one and pattern two, we use four numbers to calculate the enrichment: (1) number of grid cells in pattern 1, (2) number of grid cells in pattern 2, (3) number of grid cells in overlap, and (4) the total number of grid cells in the image (constant for each developmental stage). We then calculate the enrichment in the following way:

$$enrichment = \frac{\frac{overlap}{pattern\ 1}}{\frac{pattern\ 2}{total\ image}} \quad (4.3)$$

As an example, consider a pot containing 100 marbles ( $\hat{=}$  *total image*) in which 10 of them are black ( $\hat{=}$  *pattern 2*) and the rest white. Now you take a sample of 50 marbles ( $\hat{=}$  *pattern 1*) and count 10 black ones among them ( $\hat{=}$  *overlap*). The enrichment in this specific example would therefore be 2 since  $\frac{10/50}{10/100} = 2$ .

As another example, let's consider *Drosophila* embryos again since we are not



**Figure 4.21.:** Two staining patterns (see Figures 4.20 (a) and 4.20 (c)) and their overlap are used to calculate a fold enrichment for a given intensity threshold.

randomly drawing balls but comparing patterns. Figure 4.21 shows part of the comparison process of the pictures in Figures 4.20 (a) and 4.20 (c). By applying a threshold to the pattern intensity, we obtain binary masks showing all the pixels passing the cutoff (see Figures 4.21 (a) and 4.21 (b)). By putting the two masks on top of each other, we obtain the overlap shown in Figure 4.21 (c). Thereby, all the numbers needed for Equation 4.3 are available and we can calculate the enrichment:

$$enrichment = \frac{\frac{overlap}{pattern\ 1}}{\frac{pattern\ 2}{total\ image}} = \frac{\frac{155}{161}}{\frac{298}{472}} = 1.5249 \quad (4.4)$$

The example shown in Figure 4.21 explains how we obtain an enrichment value for a certain threshold. Additionally, we calculate a p-value for each enrichment using the cumulative hypergeometric distribution and only consider those passing a p-value cutoff of  $10^{-4}$ . Therefore, for each intensity threshold we get (1) a value for an enrichment or a depletion and (2) the significance of this enrichment or depletion. We now calculate the enrichment and significance for thresholds starting at zero and ending when one of the patterns does not have any pixels passing the corresponding threshold. See Table 4.1 for the complete range of thresholds and obtained enrichment values for the example started in Figure 4.21. As a final quantification of similarity we use the natural logarithm of the maximum enrichment obtained in the course of raising the threshold. For the example shown in Table 4.1, the final similarity value is  $\log(2.73) = 1.00$ .


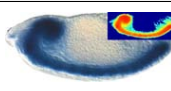
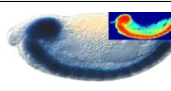
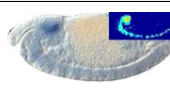

See Table 4.2 for the actual normalized enrichment values of the examples shown in Figure 4.20 at the beginning of this section. Normalization is being performed as the first step of clustering and is therefore introduced in Section 4.7.1.

#### 4.6.2. Other Approaches for Calculating Pattern Similarity

Besides the method of *mutual enrichment* introduced in this section, we also considered two other methods for image comparison: The pearson correlation

Threshold	Overlap	Pattern 1	Pattern 2	Total	Fold Change	p-Value
0	155	161	298	472	1.52	$1.58 \times 10^{-33}$
0.05	106	110	267	472	1.70	$1.00 \times 10^{-27}$
0.1	80	80	249	472	1.90	0.00
0.15	65	65	239	472	1.97	0.00
0.2	55	55	229	472	2.06	0.00
0.25	47	48	217	472	2.13	$2.71 \times 10^{-18}$
0.3	42	42	205	472	2.30	0.00
0.35	33	33	194	472	2.43	0.00
0.4	18	18	184	472	2.57	0.00
0.425	12	12	178	472	2.65	0.00
0.4375	11	11	173	472	<b>2.73</b>	0.00

**Table 4.1.:** Exemplarious pattern comparison of image shown in Figures 4.20 (a) and 4.20 (c). The maximum enrichment of 2.73 (or actually  $\log(2.73) = 1.00$ ) is the final output of the comparison shown.

				
	1	0.89	0.64	0

**Table 4.2.:** Normalized pairwise similarities of four control comparisons showing high similarity of an image with itself (column 2) or a very similar image (column 3), medium similarity to a related pattern (column 4), and no similarity to an unrelated pattern (column 5). Values shown are normalized logarithmic fold enrichments.

(1) and percent overlap (2).

The first method works by *correlating* two arrays of pattern intensities extracted out of the images to be compared using Pearson’s correlation. To illustrate this on an example, we would generate binary masks as shown in Figures 4.21 (a) and (b) out of the patterns intensities. Only the grid cells which are set in either binary mask are then considered for calculating the pearson correlation coefficient.

As a second method, we considered *percent overlap* which is, assuming we are comparing two patterns, how much of the smaller pattern is contained in the larger pattern:

$$\text{percentOverlap} = \min \left( \frac{\text{overlap}}{\text{pattern 1}}, \frac{\text{overlap}}{\text{pattern 2}} \right) \quad (4.5)$$

As with mutual enrichment, we repeat this calculation with increasing thresholds until one of the patterns is empty and use the largest value of *percentOverlap* found as the actual similarity.

We evaluated these three methods in Section 4.7 using a test set and found *pearson correlation* to be the worst and *mutual enrichment* to be slightly better than *percent overlap*. See Section 4.7.2 for details on the individual performance of each method.

We can now use the means for comparing two embryo images introduced in this section in order to look at a larger number of images and cluster them by expression pattern.

## 4.7. Clustering of Images with Similar Patterns

The previous section explained how we compute pairwise similarities. This section explains how we use these pairwise similarities to cluster images using MATLAB and how well this works on a test set.

### 4.7.1. Normalization

By calculating the logarithmic fold changes (i.e. similarities) as described in the last section, we obtain a matrix of pairwise similarities. This matrix is symmetrical along the diagonal since *A compared to B* equals *B compared to A*. In order to proceed with clustering the images, we first make sure the values of the similarity matrix are in the range  $[0, 1]$ . This is done by (1) subtracting the smallest enrichment value possible and (2) by dividing a similarity value  $sim_{i,j}$  for the similarity between the images *i* and *j* by  $min(sim_{i,i}, sim_{j,j})$ . The rationale behind the second step is to normalize each enrichment to the highest value possible for this comparison. The similarity between images *i* and *j* cannot be higher than the similarity of either *i* or *j* to itself, therefore

$$sim_{i,j} = \frac{sim_{i,j}}{min(sim_{i,i}, sim_{j,j})}.$$

The transformed similarity matrix *sim*, containing exclusively values in the range  $[0, 1]$ , is then converted to a distance matrix *dist* by applying  $dist = 1 - sim$ . This distance matrix is then subjected to the MATLAB-provided function `linkage` which creates an agglomerative hierarchical cluster tree using the unweighted pair group method with arithmetic mean (UPGMA) [58]. In UPGMA, the distance between two clusters is the average distance between the objects in either cluster. The resulting tree can then be visualized as a dendrogram (see code).

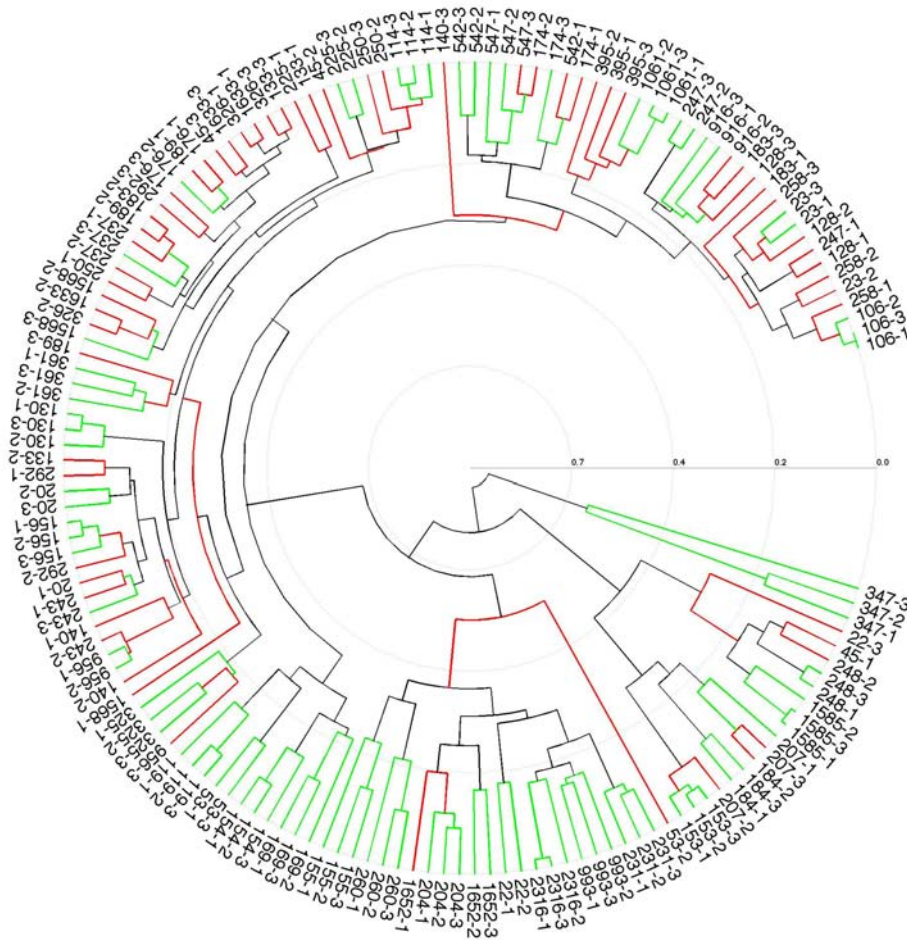
### 4.7.2. Performance

In order to evaluate our similarity quantification we picked a test set of images and systematically assessed its performance. We randomly selected 50 *Drosophila* lines and picked, again randomly, three images for each line for stages 13-16 in lateral orientation and applied our clustering pipeline using mutual enrichment as the similarity score. The resulting dendrogram is shown in Figure 4.22.

Each leaf node in Figure 4.22 corresponds to one picture and is labeled following the syntax “X-Y” where X specifies the genetic line and Y the number of the picture. The leaf node labeled 97-1 would therefore correspond to picture number one of *Drosophila* line number 97.

For assessing the performance of the clustering, the lines in the dendrogram joining the leaf nodes are colored in green or red depending on the validity of this join. If one leaf node is joined to another leaf of the same genetic line or to a cluster comprised majorly (>50%) of the same genetic line, the join is colored green (*good join*). Otherwise, the join is colored red (*bad join*). In Figure 4.22,





**Figure 4.22.:** Clustering of a test set consisting of 150 images taken from 50 randomly chosen fly lines. The clustering is visualized as a polar dendrogram where the leaf nodes (i.e. single images) are aligned in a circular fashion as opposed to a linear dendrogram in order to save space.

57% of the joins clustering single lines are green (i.e. valid). This means that in 57% of the cases leaf nodes of corresponding genetic backgrounds are joined.

Having considered single join events, we now look at the performance in terms of *Drosophila* lines. We expect from a clustering such as the one shown in Figure 4.22 that all the images of biological replicates cluster together, e.g. images 97-1, 97-2, and 97-3 form a subtree before any other image is joined. Among the 50 lines considered in Figure 4.22, for 18 lines (36%) all three images are clustered together, and for 12 lines (24%) two out of three images are clustered together. Therefore, 30 out of 50 lines (60%) can be considered as clustered correctly.

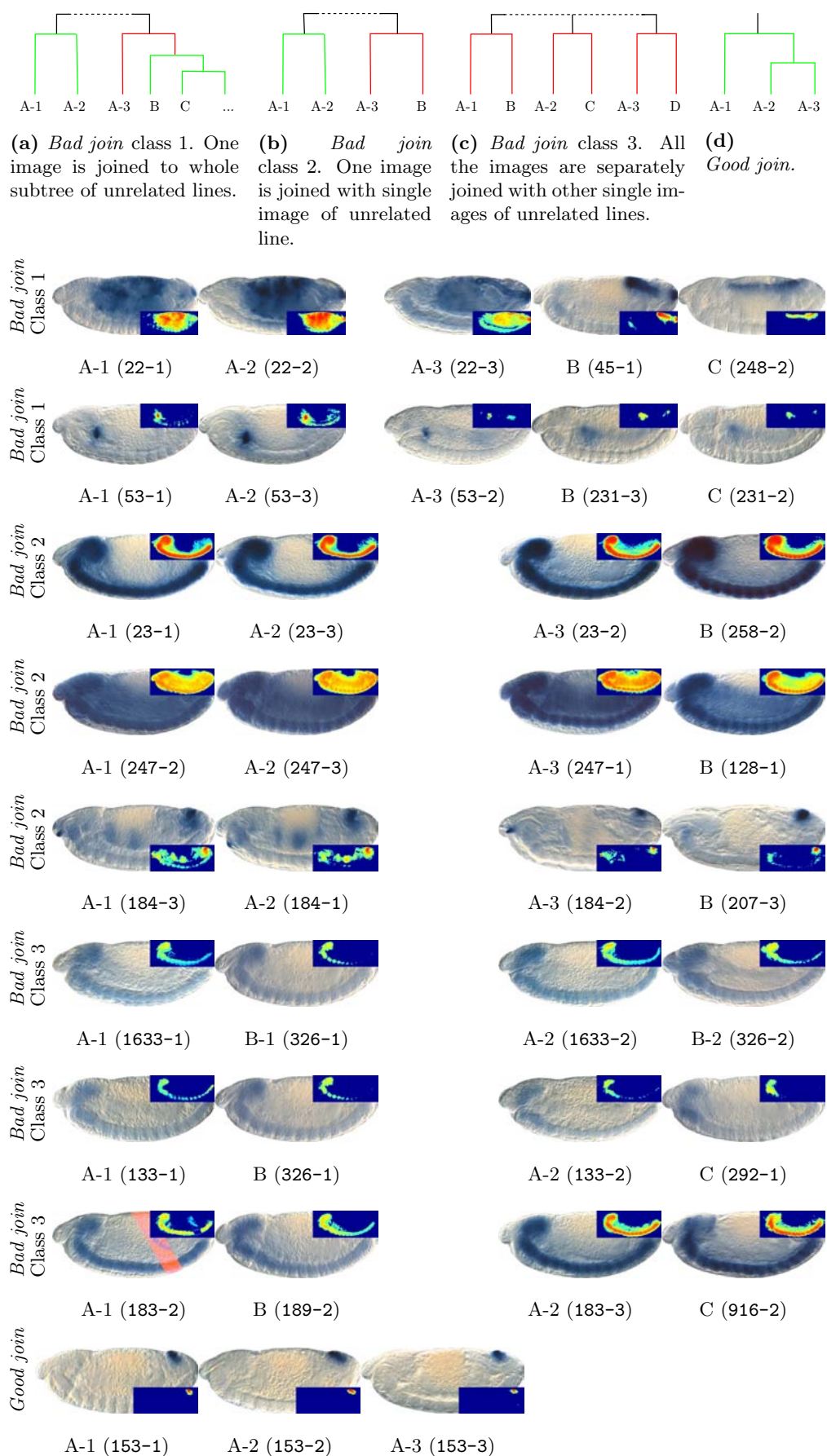
In Section 4.6, we briefly mentioned two additional methods of calculating similarity besides mutual enrichment: Pearson correlation and percent overlap.

Evaluating these two methods using the same test set as for mutual enrichment, we find for *pearson correlation*, the good-joins ratio is 48% and for 11 lines (22%) all three images are clustered together, and for 17 lines (34%) two out of three images are clustered together.

For *percent overlap*, the good-joins ratio is 43% and for 8 lines (16%) all three images are clustered together, and for 25 lines (50%) two out of three images are clustered together.

We can therefore conclude that the method based on pearson correlation performs worst with more than half of the leaf nodes being clustered wrongly and 56% of the lines clustered correctly. Percent overlap performs even worse in terms of the good-joins ratio with a value of 43% but a lot better in the correct clustering of lines with 66%. Mutual enrichment, which is the method used throughout this section, has the highest good-joins ratio with 57%. Compared to percent overlap, mutual enrichment's success in clustering lines is lower with 60% but with a far higher ratio of complete lines (i.e. 3/3 images clustered) of 36% compared to 16% with percent overlap.

Let's take a closer look at the seemingly wrong joins in the dendrogram shown in Figure 4.22 and see why some images of genetically unrelated lines have been clustered together (*bad joins*). Shown in Figure 4.23 is a representative selection of image pairs which got joined in the dendrogram in Figure 4.22 even though they represent expression patterns of unrelated lines. Figures 4.23 (a) - (c) schematically show the *major three classes* of how images of biological replicates are clustered wrongly: One image is clustered with a *whole subtree* of genetically unrelated lines (a), one image is clustered with a *single unrelated image* (b), or all three images are *separately joined* with other unrelated images (c). Additionally, Figure 4.23 (d) shows how all images of one line are clustered together before any other unrelated image joins. The major part of Figure 4.23 shows a couple of examples for each of the mentioned classes. For each example, the embryo images involved, including a small pattern heatmap in the top or bottom right corner, are shown. Each image is labeled according to its role in the schematics shown in Figures 4.23 (a) - (d) and in parentheses the image identifier as in the dendrogram in Figure 4.22. Upon closer inspection of the examples in Figure 4.23, it is clear that these images which got wrongly clustered together are actually different enhancers with similar or identical properties and thereby explain and validate the behaviour of our algorithm.



**Figure 4.23.:** Examples of *Bad joins*. Seemingly wrong joins in the dendrogram shown in Figure 4.22 are explained by their similarity in expression of genetically unrelated *Drosophila* lines.

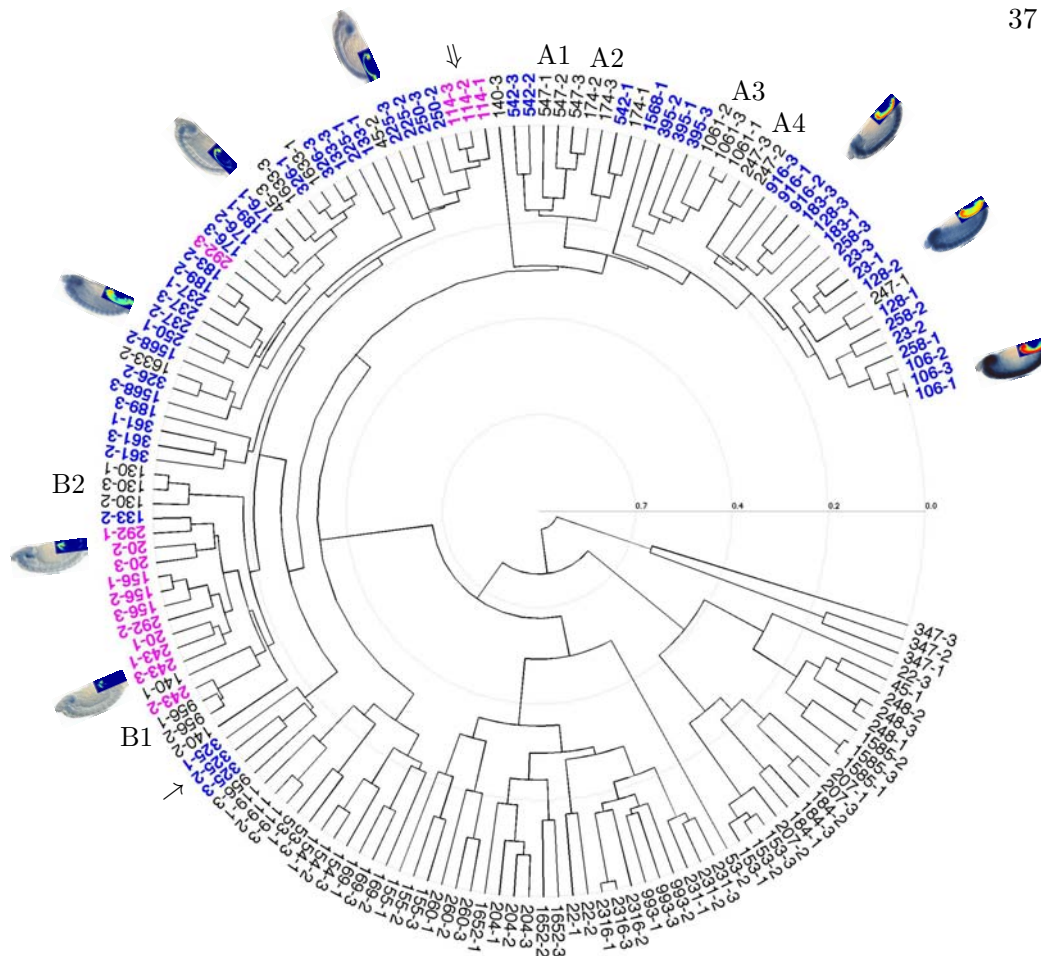
As a final performance consideration, we analyze how well the automatic clustering conforms to manual annotations. In the course of this study, we constructed a test set by manually selecting images of the most prominent classes of enhancer activity. This was done independently of the automatic clustering described in this chapter. The expression patterns of our reporter gene which we found most frequently are related to the nervous system: The whole developing central nervous system, termed CNS, (1) and the embryonic brain only without involvement of the ventral nerve cord (2). See Figure 4.24 for an overview of the clustering with the classes CNS/brain marked and a couple of images corresponding to special cases.

The clustering shown in Figure 4.24 (a) is the same as the one in Figure 4.22. In Figure 4.24 (a), the image labels are additionally stained in blue and magenta corresponding to manual annotations: *Drosophila* lines annotated as being active in the whole CNS are stained *blue* (1) and lines active in the brain only are stained *magenta* (2). The dendrogram in Figure 4.24 (a) shows that the manually annotated images for CNS are largely clustered together, and the same is true for the lines active in the embryonic brain. The images of these two classes are in close vicinity due to their apparent similarity or overlap in expression but distinct enough from each other to have the members of each class clustered together without joining the other class.

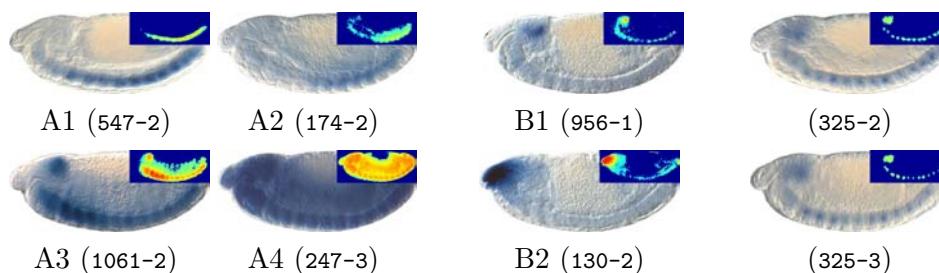
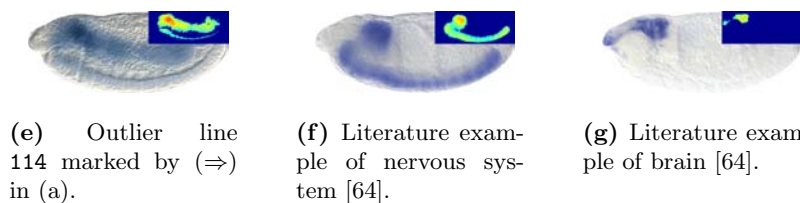
In a few exceptional cases shown, images annotated as CNS are in a brain cluster (line 325, marked by  $(\rightarrow)$  in Figure 4.24 (d)) or the other way around (line 114, marked by  $(\Rightarrow)$  in Figure 4.24 (e)).

Additionally, as shown in Figure 4.24 (a), some not-annotated lines clearly cluster together with images manually annotated as CNS or brain. Figure 4.24 (b) shows four representative images of lines which are inside a larger CNS cluster but not annotated as such. Figure 4.24 (c) shows two representative images of lines which are inside a larger brain cluster but not annotated as such. The images shown validate our clustering algorithm due their high similarity to the classes CNS/brain; even though a human expert might judge the tissues as being different, the visual resemblance, which is what our clustering algorithm is trying to assess, is given. Additionally shown for comparison are images of *in-situ* hybridizations for the genes *Olig family* (Figure 4.24 (f)) and *Optix* (Figure 4.24 (g)) which are expressed in the CNS and the brain respectively. The images shown in Figures 4.24 (f) and (g) underwent the cropping and pattern extraction algorithm described in this chapter, the raw images are taken from the Berkley Drosophila Genome Project (BDGP) website [64].

To conclude, we can assume from the data presented in this chapter, especially in Figures 4.22, 4.23, and 4.24, that the algorithms developed for the segmentation, orientation, pattern extraction, registration, and clustering of *Drosophila* embryo images perform well and provide the means to identify similarities in expression.



(a) Dendrogram.

(b) Manually not annotated as but clustered with *nervous system* lines.(c) Manually not annotated as but clustered with *brain only* lines. (d) Outlier line 325 marked by ( $\rightarrow$ ) in (a).(e) Outlier line 114 marked by ( $\Rightarrow$ ) in (a).

(f) Literature example of nervous system [64].

(g) Literature example of brain [64].

**Figure 4.24.:** Automatic clustering of images by expression pattern fits independent manual annotations. (a) shows the same clustering as Figure 4.22 and, in addition, manual classifications: Central nervous system (CNS; *blue*) and brain only (*magenta*). (b) and (c) show examples of images clustered with CNS/brain but lack the corresponding manual annotation. (d) and (e) show one line each of CNS/brain which got clustered in the respective other class. (d) and (e) are examples of CNS/brain expression from the literature [64].

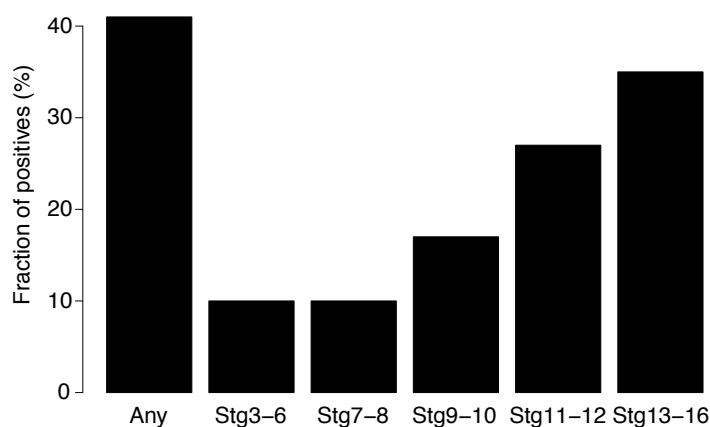


## 5. Results on Enhancer Activity

Using the approach established here, we find more than 44% of the elements tested to be active in *Drosophila* embryogenesis. In this chapter, we first report on the positive rate found in the various stages of development and then analyze the major groups of found enhancers using the computational tools introduced in Chapter 4. Additionally, we show a couple of genomic loci containing enhancers we found and analyze their putative regulatory influence on adjacent genes.

### 5.1. Positive Rate

In this section we report the fraction of positive *Drosophila* lines, i.e. flies carrying a DNA fragment which is able to activate transcription of our reporter gene *Gal4*. We found 403 out of the 981 enhancers candidates examined to be active in at least one stage of embryogenesis. We also observe a rising positive rate with progressing development as shown in Figure 5.1: Of 981 candidates examined, 97 (10%) are active in *stages 3-6*, 99 (10%) in *stages 7-8*, 171 (17%) in *stages 9-10*, 265 (27%) in *stages 11-12*, and 342 (35%) in *stages 13-16*. Overall, 403 (41%) elements are active in at least one stage.



**Figure 5.1.:** The fraction of enhancer candidates being active increases with progressing development.

The trend of a rising positive rate with progressing development is consistent with the increasing complexity of the embryo in later stages which requires more complex gene regulation. This is therefore also reflected in the number of active genes during embryogenesis. In a study based on *in-situ* expression data [64], the fraction of genes expressed in the various stages of embryogenesis (excluding maternally loaded transcripts) was found to be increasing over time:

7087 genes examined, 2533 (36%) active in *stages 4-6*, 2835 (40%) in *stages 7-8*, 3033 (43%) in *stages 9-10*, 3539 (50%) in *stages 11-12*, and 4397 (62%) in *stages 13-16*.

The increasing tissue complexity in the course of embryogenesis can be also observed by looking at the annotations of the genes' expression patterns in [64]. Each of the genes examined by *in-situ* hybridization is associated with manual annotations for each time point in development. Considering the number of unique terms used in each stage and over all genes, we again see an increasing trend: Three unique terms in stages 1-3, 9 in stages 4-6, 14 in stages 7-8, 24 in stages 9-10, 70 in stages 11-12, and 105 in stages 13-16.

## 5.2. Spatio-Temporal Activity

We used the image clustering algorithm described in Section 4.7 in order to identify commonly occurring patterns.

### 5.2.1. Prominent Patterns

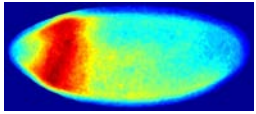
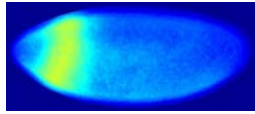
The most prominent group we found are enhancers active in the developing nervous system which includes the embryonic brain and the ventral nerve cord (VNC). Figure 5.2 shows the clusters we identified and the location of the *in-situ* signal for each cluster in the columns *pattern presence* and *pattern intensity*.

For *early* development (stages 3-6), with a low positive rate compared to the late stages, we found the most often occurring pattern to be in an anterior region of the blastocyste called procephalic ectoderm as shown in Figure 5.2 (a).

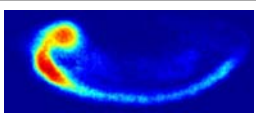
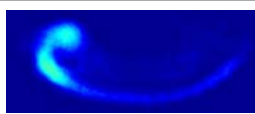
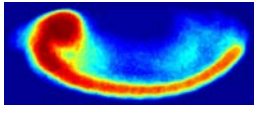
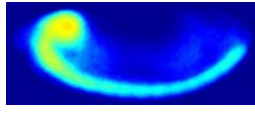
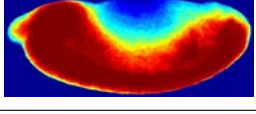
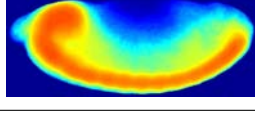
For the *late* time points in development, stages 13-16, we found three major subclasses of nervous system enhancers shown in Figure 5.2 (b). *Cluster two* is majorly active in the developing brain only. *Cluster three* is active in the brain and in the VNC and *cluster four* additionally shows ubiquitous expression.

Figure 5.2 shows four clusters, each containing enhancers able to give rise in similar regions of the early or late *Drosophila* embryo. Next, we are having a look if these lines also show similarities in their behaviour over time.

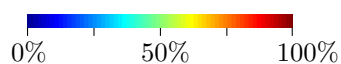


Cluster	Pattern presence	Pattern intensity	Description
1			Procephalic ectoderm (80 images, 37 lines)

(a) Early development (stages 3-6)

Cluster	Pattern presence	Pattern intensity	Description
2			Brain only (91 images, 43 lines)
3			CNS (114 images, 49 lines)
4			CNS, ubiquitous (78 images, 24 lines)

(b) Late development (stages 13-16)



**Figure 5.2.:** Most prominent clusters found through image analysis show activity in the developing nervous system. **(a)** shows the prominent cluster for early development in stages 3-6. The area showing the most expression here is the procephalic ectoderm which is where the head of the fly will develop. **(b)** shows the three major clusters identified for the late developmental stages 13-16: Brain only (2), brain and VNC (3), and brain, VNC, and ubiquitous (4). The heatmaps shown for *pattern presence* provide binary information on how many of the images in this cluster show any expression (regardless of the intensity) in the corresponding pixel. *Pattern intensity* shows the average intensity over all images of a cluster. Additionally shown for each cluster is the number used for reference in the text (column 1) and a short description (column 4).

### 5.2.2. Temporal Dynamics

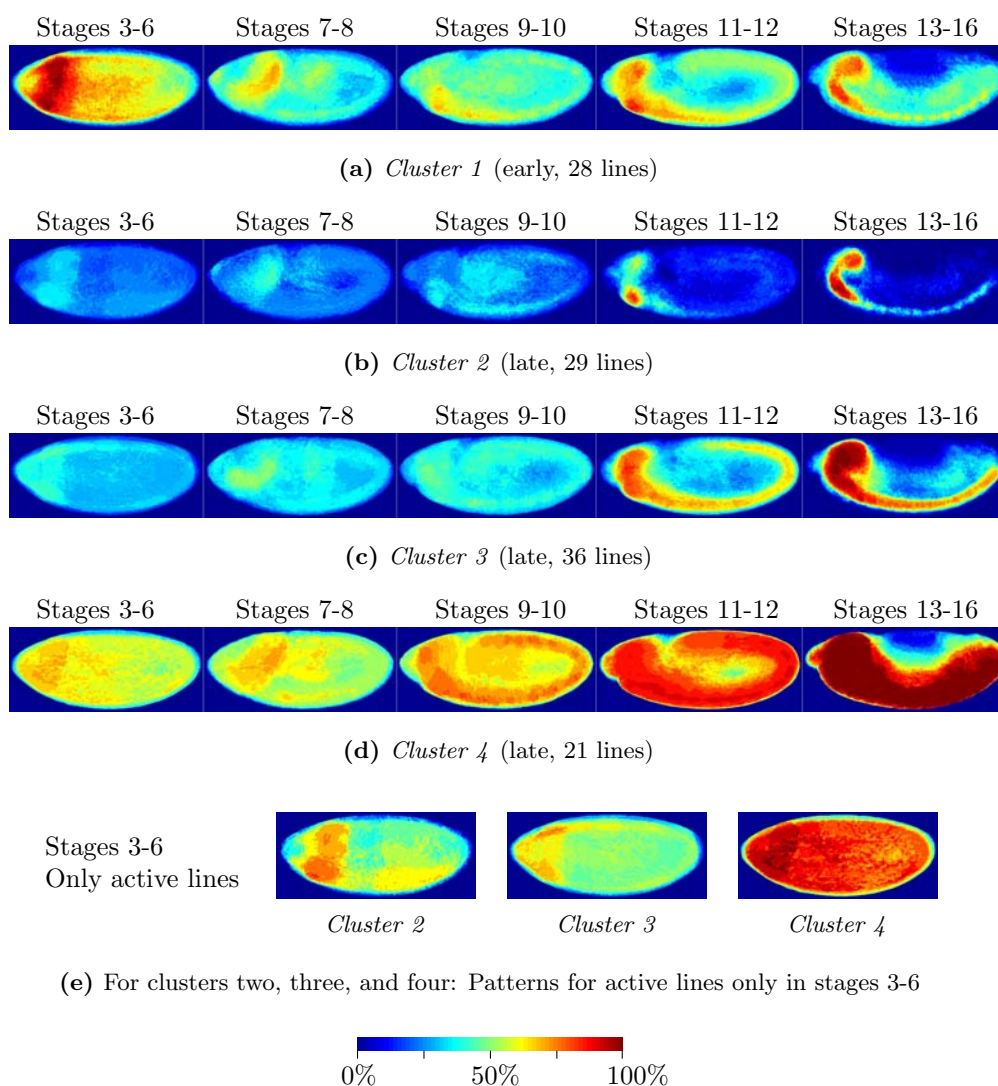
In the previous section, we grouped the enhancers found in this study into major groups of spatial activity for early (stages 3-6) and late (stages 13-16) *Drosophila* embryogenesis.

For *early stages*, we found that the corresponding enhancers (*cluster one*) keep their activity throughout development: Almost all of the enhancers active in the procephalic ectoderm at early stages remain active in the embryonic brain or CNS at later stages (see Figure 5.3 (a)).

For *late stages*, it seems that the temporal relationship early  $\rightarrow$  late is not true for the reverse direction late  $\rightarrow$  early. Whereas more than 70% of the lines active in the procephalic ectoderm at early stages are turned on in the brain in late stages (*cluster one*), the majority (59%) of lines active only in the embryonic brain in late stages (*cluster two*) are completely turned off in early stages (see Figure 5.3 (b)). Also, 47% of the lines in *cluster three* are turned off in early stages (see Figure 5.3 (c)). However, looking only at the remaining lines which are already active in stage 3-6 in Figure 5.3 (e), it seems that there is a trend towards the procephalic ectoderm for the late clusters. In general, the majority of active stage 3-6 enhancers in clusters two, three, and four show at least partial activity in the region of the procephalic ectoderm: Cluster two 83%, cluster three 58%, and cluster four 50%.

Additionally, when looking at the activity of *cluster four* in early stages 3-6 in Figure 5.3 (d), it seems to contain a higher fraction of lines active in the procephalic ectoderm compared to clusters two and three. This is due to the fact that enhancers active ubiquitously in late stages tend to be active ubiquitously in all the other stages too. Moreover, the ubiquitous activity observed in cluster four is not an artifact caused by overstaining strong enhancers active in the brain or VNC. We can exclude this since we observe (1) lines with a strong activity in the brain and VNC without any ubiquitous expression and (2) lines active in the brain, VNC, and ubiquitously where the brain- and VNC-part is weaker than some lines active in the brain and VNC only. It is therefore clear that the additional ubiquitous component in cluster four is qualitatively different from strong CNS lines.

To conclude, the spatio-temporal dynamics observed in Figure 5.3 are in accordance with the tissue fates in *Drosophila* embryogenesis. We observe that enhancers active in neurogenic regions in the very early embryo remain active in central nervous system tissue in the late stages of development. It is known that most of the cells forming the procephalic ectoderm of the early embryo delaminate and move inside the embryo to give rise to neuroblasts which will later form the larval brain [29]. The other way around is not true: In fact, the majority of enhancers active only in the brain late, for example, are *off* entirely early (*cluster two*). Of the ones already active early, however, the majority is active in the region of the procephalic ectoderm as shown in Figure 5.3 (e). Overall, around  $\frac{1}{3}$  of the enhancers active in the CNS in late stages is active in the procephalic ectoderm in early stages.



**Figure 5.3.:** Temporal development of pattern clusters introduced in Section 5.2. (a) Cluster 1: The majority of enhancers active in the procephalic ectoderm in stages 3-6 are later on, in stages 13-16, active in the embryonic brain and to a lesser degree in the VNC. (b) - (d) Clusters 2-4 were identified as being similar in their activity in stages 13-16 and seemingly derive from procephalic regions in stage 3-6. (e) Same as in (b) - (d) for stages 3-6, except that only active lines are shown.

Each pixel in the heatmaps is stained corresponding to the fraction of lines active in this region, ranging from 0% (dark blue) to 100% (dark red).

### 5.3. Genes - The BDGP Dataset

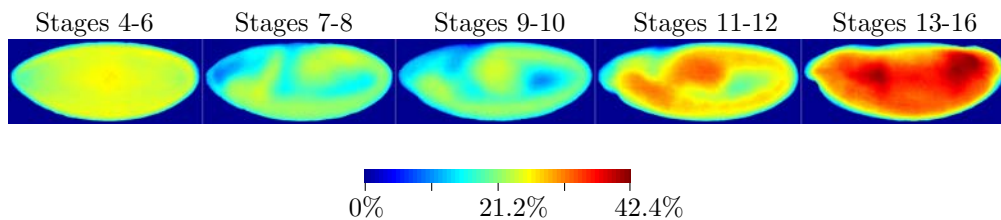
We also applied our computational pipeline for image segmentation, pattern extraction, etc. (see Section 4) to the Berkeley Drosophila Genome Project (BDGP) dataset of *in-situ* pictures [64]. This dataset contains 82 699 manually annotated images of *in-situ* hybridizations to 6 936 transcripts in the *Drosophila*

embryo. The annotations for each image include the stage of embryogenesis, the orientation of the embryo, and the stained tissues using a manually curated vocabulary.

At the time of writing (November 2010) the BDGP *in-situ* database snapshot<sup>1</sup> contained annotations and image URLs for 6 936 genes. Of these 6 936, 3 906 genes are annotated as being expressed in at least one of the stages 4-6, 7-8, 9-10, 11-12, or 13-16. We were able to obtain the images for 3 814 of these genes and applied our computational pipeline. The image segmentation step introduced in Section 4.2 was successful for 3 603 (94%) genes. However, of these 94% around one third had to be sorted because of quality reasons. After manually sorting out images of low quality due to light, focal conditions, or aberrantly segmented embryos, we ended up with a set of 2 356 (62%) genes (the *2.4k set* of genes). For each of these genes we therefore now have at least one image for every stage the gene is expressed in according to the BDGP.

Using the 2.4k set as a representative sample of all genes active in embryogenesis, we analyzed the spatial distribution of gene expression for each of the developmental stages (see Figure 5.4). In order to create this pattern presence heatmap for a specific stage, we considered one image per gene. This image, specific for a certain gene and stage, might be either the average over all the images available for this gene and stage or an empty image (containing zeros only) if the gene is annotated as inactive in this stage. Additionally, we excluded expression patterns annotated as “maternal” by the BDGP curators for all subsequent analysis of the 2.4k set since we are only interested in active transcription.

Figure 5.4 shows that, for the 2.4k set at least, gene expression is not restricted to any particular location in the *Drosophila* blastoderm in stages 4-6. Later on, in stages 7-8, 9-10, 11-12, however, active genes are concentrated in cells which are part of the germ band. In the late stages 13-16, 62% of the genes examined by the BDGP are active and expressed throughout the whole embryo. Virtually the same patterns as the ones in Figure 5.4 were found by a previous approach also based on the BDGP dataset [34].



**Figure 5.4.:** *In-situ* pattern presence of 2 356 *Drosophila* genes among different developmental stages of embryogenesis. Images annotated as showing maternally loaded transcripts are excluded, only zygotic transcripts are considered. Raw images were obtained from [64], subjected to our computational pipeline, and the resulting images were manually inspected and sorted for quality.

Next, we used the 2.4k set again to ask whether the temporal dynamics we

<sup>1</sup><http://insitu.fruitfly.org/insitu-mysql-dump/insitu.sql.gz>

found for enhancers in Section 5.2.2 are also true for genes. For this purpose, we made use of the rich vocabulary used to annotate the BDGP images. We again created pattern presence heatmaps in the same manner as shown in Figure 5.4, this time with an input set restricted to genes which are annotated as expressed in (1) the *procephalic ectoderm* in stages 4-6 (see Figure 5.5 (a)) and (2) the *ventral nerve cord* or *brain* in stages 13-16 (see Figure 5.5 (b)).

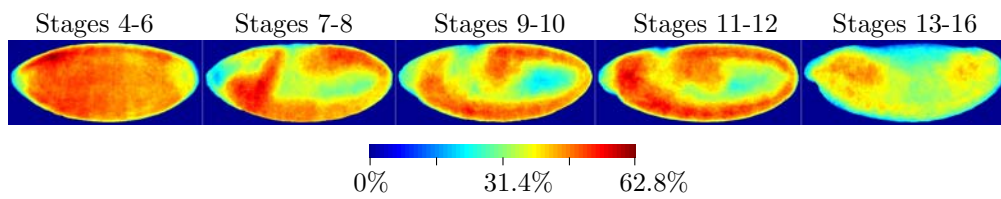
As shown in Figure 5.5 (a), genes expressed in the *procephalic ectoderm in early stages* (as determined by the BDGP annotations) seem to be expressed in the region of the brain and central nervous system in late stages 13-16, as judged by visual inspection. Comparing this temporal dynamics of genes to the dynamics of enhancers (Figure 5.5 (a) versus Figure 5.3 (a)), it is clear that the regions of procephalic ectoderm in early stages and the brain/VNC in late stages are not as specifically pronounced for genes as for enhancers. This might be caused by technical and biological factors:

*Technically*, the BDGP dataset for late stages is more heterogeneous than our data. While the BDGP images annotated as stages 13-16 actually document embryos in all of these stages, we only took images of stage 13 embryos as representative of the stages 13-16. Therefore, due to major morphological movement in the head area in between stages 13 and 16, the brain in stages 13-16 in Figure 5.5 (a) is not as enriched with *in-situ* signal as we observe it for enhancers.

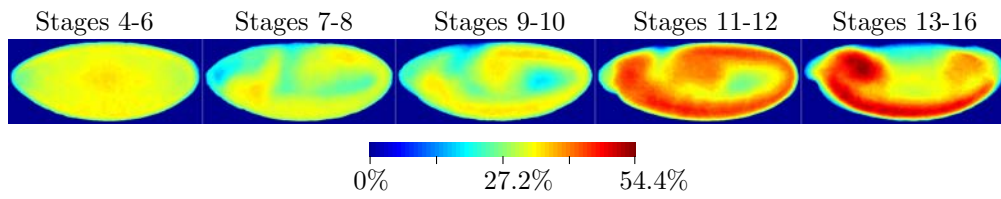
Possible *biological* explanations include the assumption that enhancers are modular. In Figure 5.5 (a), we observe that the expression pattern of genes manually annotated as being expressed in the procephalic ectoderm in early stages is actually spread all over the embryo. Excluding the possibility of mis-annotations, this means that expression in this early ectodermal stripe rarely appears alone but goes together with other regions in the embryo. On the other hand, we do observe enhancers specifically active only in the procephalic ectoderm (see Figure 5.3 (a)).

For genes expressed in the *central nervous system in late stages* however (see Figure 5.5 (b)), we do not find any preferred area of expression in early stages 4-6. Again comparing this to enhancers (clusters two to four in Figure 5.3) where we see only  $\frac{1}{3}$  of the corresponding enhancers active in the procephalic ectoderm, it is possible that the biological and technical factors introduced above mask an already weak enrichment in the region of the procephalic ectoderm.

In addition to a comparison of enhancer and gene expression patterns on an aggregated level as it was done in this section, we can use the BDGP resource to focus on single loci as shown in Section 5.4.



(a) Genes annotated as *procephalic ectoderm* in stages 4-6



(b) Genes annotated as *ventral nerve cord* or *brain* in stages 13-16

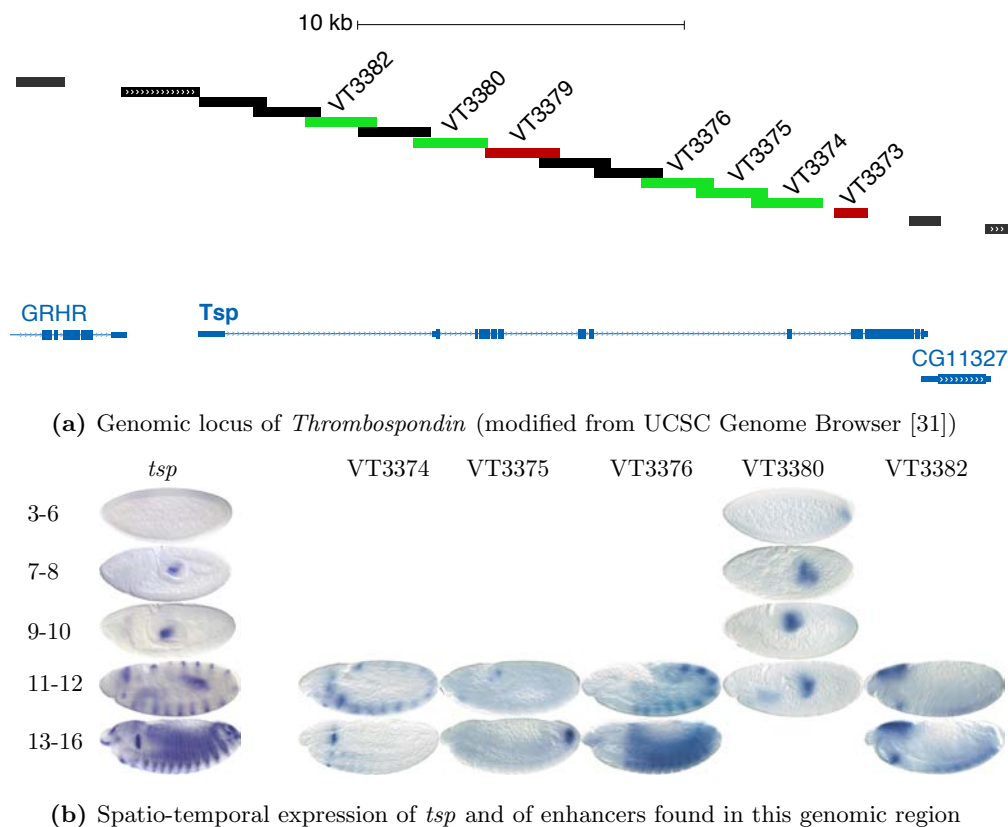
**Figure 5.5.:** Temporal development of expression for two subsets of genes. (a) Pattern presence for genes annotated as expressed in *procephalic ectoderm* in stages 4-6 (b) Pattern presence for genes annotated as expressed in *ventral nerve cord* or *brain* in stages 13-16.

## 5.4. Spatio-Temporal Additivity

Enhancers contribute to the expression patterns of genes and each gene might receive regulatory input from multiple enhancers. We can therefore zoom in on a few regions covered well in our initial screen to see how the found enhancers contribute to the expression of the adjacent genes. For information about the spatio-temporal expression of *Drosophila* genes we utilized images from the BDGP *in-situ* database introduced in Section 5.3.

### 5.4.1. Thrombospondin Locus

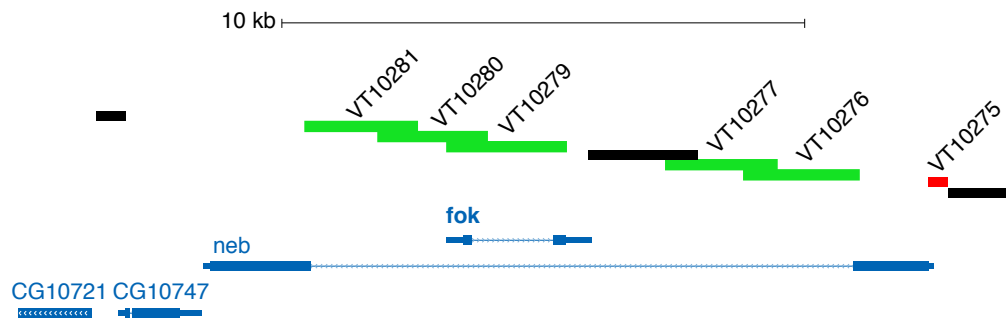
The gene *thrombospondin* (*tsp*) encodes an extracellular matrix protein and is involved in muscle-tendon attachment [1]. We screened seven regions in the *tsp* locus for regulatory activity and found five of them to be active in embryogenesis as shown in Figure 5.6. At stages 3-10, only the element VT3380 is active and recapitulates the expression of the *tsp*. Starting from stages 11-12, the remaining four elements become active whereas VT3380 soon turns off.



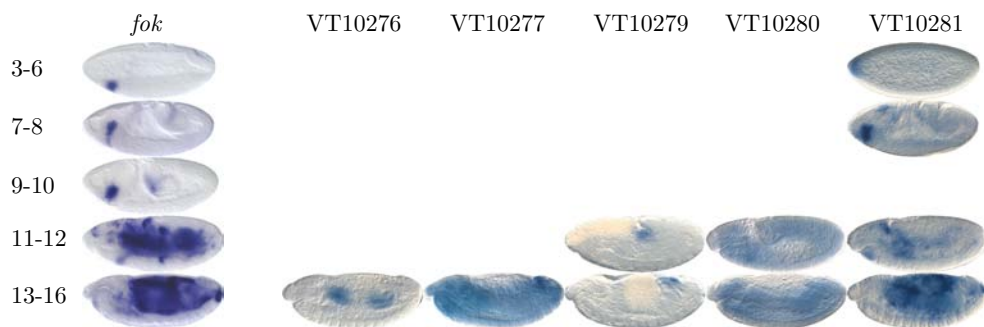
**Figure 5.6.:** Regulatory elements found at *thrombospondin* (*tsp*) locus.

### 5.4.2. Fledgling of *Klp38B* (*fok*) Locus

*Fledgling of Klp38B* (*fok*) is a gene of unknown function and located inside an intron of the gene *nebbish* (*neb*, the protein is called Klp38B) [44] which is implicated in interactions between the chromosome arms and microtubuli [41]. We found five active enhancers in the intron of *neb* where also *fok* resides (see Figure 5.7). VT10281 is highly likely the cis-element responsible for *fok* activation, since it closely resembles its expression pattern. Unfortunately, the BDGP does not hold expression information for *neb*. It is however known to be expressed in the ventral nerve cord and in procephalic regions during embryogenesis [41]. Therefore, the other four active enhancers, which do not show any activity in the nervous system, might also be contributing to the expression pattern of *fok*. Additionally, they might be contributing to other adjacent genes ever further away such as *CG10747* or *CG10721* for which no expression data is available at the BDGP either.



(a) Genomic locus of *fledgling of Klp38B* (modified from UCSC Genome Browser [31])



(b) Spatio-temporal expression of *fok* and of enhancers found in this genomic region

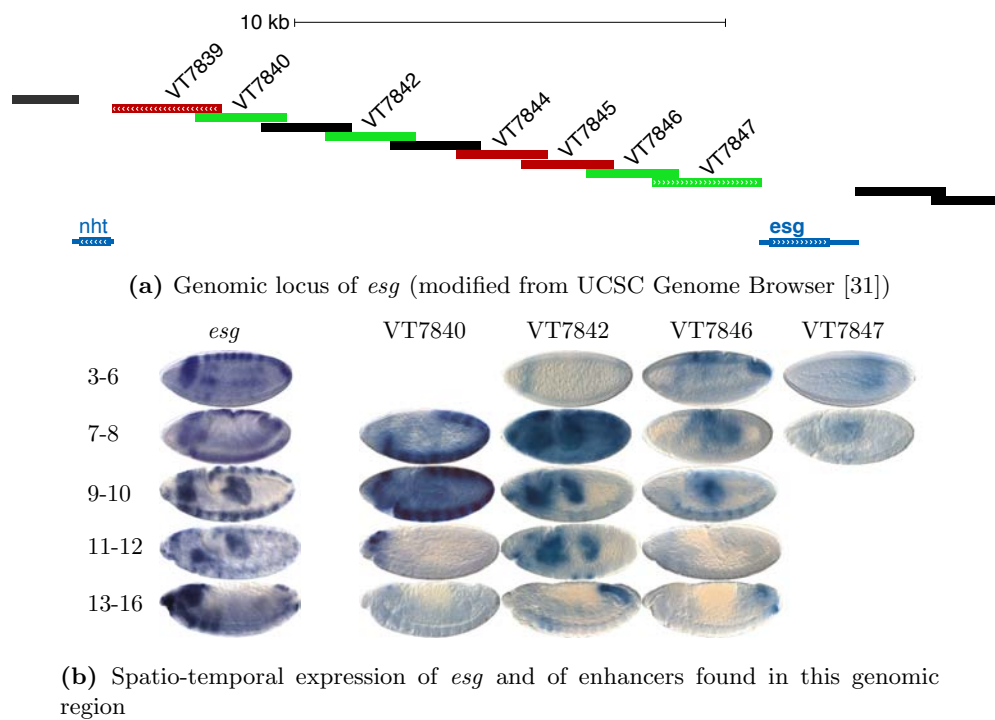
**Figure 5.7.:** Regulatory elements found at *Fledgling of Klp38B* (*fok*) and *nebbish* (*neb*) loci.



### 5.4.3. Intergenic Region between No Hitter (*nht*) and Escargot (*esg*)

The *no hitter* (*nht*) and *escargot* (*esg*) are transcribed in opposite directions and separated by a region of  $\approx 15\text{kb}$  which we investigated for regulatory activity. *Nht* is involved in the transcriptional regulation of spermatid differentiation [30], whereas *esg* has been found to play a role in tracheal [63] and nervous system development [4]. Unfortunately, BDGP does not have expression data for *nht* but only for *esg*.

We found three enhancers (see Figure 5.8), VT7840, VT7842, and VT7846, to closely resemble the *escargot* expression pattern. Interestingly, we found the 2.5kb fragment directly upstream of the *esg* transcription start site, VT7847, to mimic the expression of *esg* the least, of the four active elements found in these region. Additionally, VT7847 is active only in stages 3-6 and 7-8 and is then turned off.



**Figure 5.8.:** Regulatory elements found in intergenic region between *no hitter* (*nht*) and *escargot* (*esg*).

Summarizing this chapter, we showed that using our systematic approach, we are able to identify embryonic regulatory elements in the genome of *Drosophila melanogaster*. The fraction of active elements is increasing over developmental time which goes along well with the fraction of genes expressed and the increasing complexity of the embryo. Additionally, the computational methods introduced in Chapter 4 (1) enable us to identify similar enhancers and to observe their activity over time and (2) can be successfully applied to foreign resources such as the BDGP data set allowing us to connect enhancers to their putative target genes.

## 6. Discussion

This thesis describes a systematic approach for finding developmental enhancers in *Drosophila melanogaster*. Enhancers are the elements responsible for the spatio-temporal specificity of gene transcription and therefore crucial in the specification of different cell-types in a complex organism. Identifying enhancers in a genome, however, is not an easy task. While we are able to locate protein-coding genes in a given DNA sequence using well-defined rules known as the *genetic code*, the same is not possible for enhancers yet. Genes code for proteins in a linear fashion. In contrast, enhancers provide binding sites for transcription factors. Due to the low number of enhancers studied in the same system, it is unclear if general rules exist for the number or arrangement of these binding sites. A large data set comprising experimentally tested enhancers and their spatio-temporal activity would therefore increase our understanding of how sequence confers activity, i.e. the *regulatory code*.

We established a large-scale screen for identifying transcriptional enhancers in the model organism *Drosophila melanogaster*. In order to do this, we test fragments of the non-coding genome using *in-vivo* reporter assays. As a functional readout, we detect the transcript of a reporter gene using *in-situ* hybridizations in five different stages of embryonic development.

Our analysis reveals that 41% of the 981 tested elements are positive in at least one of the time points. Additionally, we observe an increasing fraction of active elements with progressing development. These results are consistent with an increasing number of expressed genes and tissue-complexity as determined by manual annotations [64].

In order to further analyze the activity of the enhancers found in more detail, we developed a computational pipeline for processing *in-situ* images. This allows us to automatically identify *Drosophila* embryos and quantify the *in-situ* signal in a given image. Additionally, we are able to assay the similarity of pairs of embryos in terms of their staining pattern, i.e. the enhancer activity.

Due to a bias of the library used towards the nervous system, the major class of similar enhancers is active in the developing central nervous system (CNS). A large fraction of the CNS enhancers in early stages of embryogenesis remains active in the nervous system throughout embryogenesis. This is consistent with known tissue fates [29]. In contrast, of the enhancers active in the CNS in late stages, only  $\frac{1}{3}$  is active in the developing nervous system in early stages while the majority of the remaining  $\frac{2}{3}$  is inactive in early development. The reason for this late onset of regulatory activity might be related to the proliferation of neuroblasts happening in stages 9-13. Additionally, neuronal differentiation starts in stage 13 which is likely mediated by the increased cis-regulation we observe [29]

We are also able to validate our results for several genomic loci by incorporating gene expression data. We find several enhancers to closely resemble the adjacent genes' expression patterns for loci well-covered by our tiles. For this purpose, we successfully applied our computational pipeline to a resource of *in-situ* images for genes [64]. Our results confirm the modular nature of enhancers, which we observe contributing additively to adjacent genes' expression patterns.

Additionally, our data provides the basis for assessing binding functionality of chromatin components. Current methods for profiling DNA-associated proteins, such as chromatin immunoprecipitation (ChIP) for example, reveal hundreds to thousands of regions in the genome bound by a certain transcription factor or histone variant [36]. Our data provides a functional read-out for this binding data and enables us to correlate the type, number, and arrangement of bound proteins with regulatory activity.

To conclude, we developed a large-scale method for identifying enhancers and the computational means to analyze their spatio-temporal activity. The results of around a thousand tested genomic fragments are in accordance with published literature concerning cell fates, gene expression, and tissue complexity. Ultimately, data generated by systematic screens including the one described here, will improve our understanding of enhancers and eventually lead us to the point where we can "read" a gene's spatio-temporal regulation just from the DNA sequence.

## A. Materials and Methods

This chapter lists the detailed composition or commercial sources of the materials (Section A.1) and explains the generation of the RNA probe for *in-situ* hybridization (Section A.2).

### A.1. Solutions and Commercial Reagents

**AP buffer (15ml)** 0.3ml 5M NaCl, 0.75ml 1M MgCl<sub>2</sub>, 1.875ml 0.8M Tris pH 9.5, 13.6 $\mu$ l Tween-20

**Fixative solution (15ml)** 7.5ml RNase free 4% Formaldehyde in 1xPBS (pH=7.0), 15 $\mu$ l 1M MgSO<sub>4</sub>, 15 $\mu$ l 1M EGTA, 7.5ml Heptane

**PBT** 1x PBS, 0.1% Tween-20

**PBTB** 1x PBT, 1% milk powder

**RNA hybridization solution (50ml)** 25ml formamide, 12.5ml 20x SSC, 5mg Heparine, 5mg ssSalmon Sperm DNA, 45.25 $\mu$ l Tween-20, sterile filter, aliquot, -20°C

**Anti-DIG AP antibody** Anti-Digoxigenin-AP Fab fragments Roche cat. no. 11093274910 150 U (200  $\mu$ l)

**BM Purple** Roche cat. no. 11 442 074 001 (100ml)

**DIG labelling kit** DIG RNA Labeling Mix Roche cat. no. 11 277 073 910

**NBT/BCIP** NBT (4-Nitro blue tetrazolium chloride, solution) Roche cat. no. 11383213001 3 ml (300 mg), BCIP (4-toluidine salt) Roche cat. no. 11383221001 3 ml (150 mg)

**T7 polymerase** T7 RNA Polymerase Fermentas cat. no. EP0111

### A.2. In-Situ Probe Generation

For the generation of the RNA probes against the *Gal4* transcript, we decided to PCR-amplify three different parts of the *Gal4* cDNA with primers containing the T7 RNA Polymerase promoter as 5' overhangs. As the three candidates, we chose the following regions and primers:

- Region A: Primers EK1, EK4 (1.4kb, 5' half of *Gal4*)
- Region B: EK3, EK2 (1.1kb, 3' half of *Gal4*)

- Region C: EK1, EK2 (2.5kb, full-length of *Gal4*)

As shown in Figures A.1 (d)-(f), the RNA probe spanning the whole *Gal4* gene (region C) worked best for us.

### A.2.1. PCR Primer Sequences

Primers used for *Gal4* probe generation explained in Section A.2. Sequences are shown 5' to 3'. Underlined parts are complementary to *Gal4* gene, the non-underlined sequence in EK2 and EK4 is the T7 (RNA polymerase) promoter.

**EK1** tgc gat att tgc cga ctt a

**EK2** tgt aat acg act cac tat agg gaa cat ccc tgt agt gat tcc a

**EK3** cca ccg ctc taa cca att

**EK4** tgt aat acg act cac tat agg gaa ttg gtt aga gcg gtg g

### A.2.2. Protocol

The RNA probes are named accordingly to the Regions: *Probe A*, *Probe B*, and *Probe C*. We first did a gradient PCR (94° melting, 48-59° annealing, 72° extension, 35 cycles) in order to find the optimal annealing conditions.

We started getting the correctly sized product at the end of the gradient (59°, data not shown), so we redid the gradient PCR with the annealing temperatures shifted upwards (94° melting, 57-65° annealing, 72° extension, 35 cycles). The results are of this second gradient PCR are shown in Figure A.1 (a).

The PCR products shown in lanes one to three of Figure A.1 (a) were cut-out and purified (each solved in 50µl nuclease-free water). This resulted in the following yields:

- Region A: 44.84ngµl
- Region B: 81.27ngµl
- Region C: 59.65ngµl

See Figure A.1a (b) for a gel of the pooled and purified products.

We continued with the in-vitro transcription of the probe using the DIG RNA labeling mix from Roche. Following the kit instructions we set up a 25µl reaction mix:

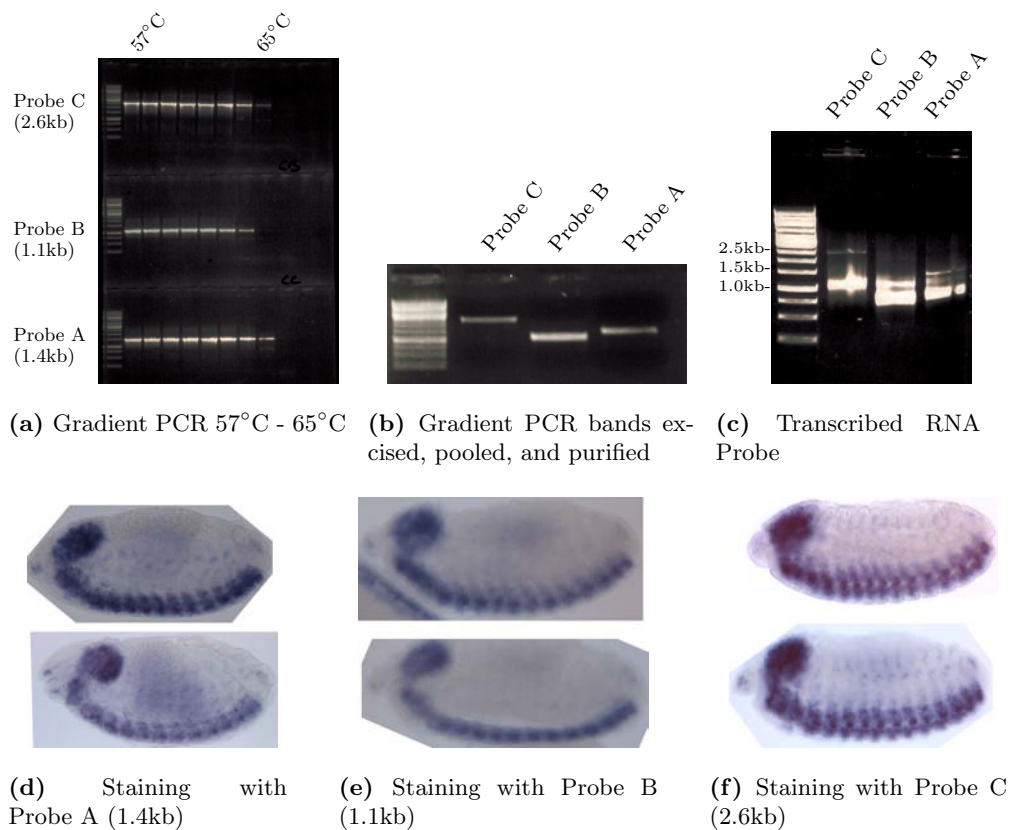
- 500ng sample DNA (PCR product)
- 2.5µl DIG labeling mix
- 5µl 5x buffer
- 2.5µl T7 RNA polymerase

- filled with nuclease-free water to 25 $\mu$ l

After the reaction we put 1 $\mu$ l on a control gel, filled the tube up to 50 $\mu$ l with water and precipitated the probe using Na-Acetate and Ethanol overnight at -80°. The next day we dried and resolved the DNA, RNA-mixture in 21 $\mu$ l H<sub>2</sub>O and put 1 $\mu$ l onto a control gel (see Figure A.1 (c)).

Overall, this protocol yielded the following RNA concentrations:

- Region A: 1.28 $\mu$ g $\mu$ l
- Region B: 1.18 $\mu$ g $\mu$ l
- Region C: 1.21 $\mu$ g $\mu$ l



**Figure A.1.:** *In-situ* RNA probe generation and staining results. (a)-(b) PCR from cDNA (c) RNA probe *in-vitro* transcribed from PCR product. Each lane shows an upper band (DNA template) and a lower band (transcribed RNA) (d)-(f) Stainings of *elav*-driven *Gal4* performed with generated probes. Only probe C shows peripheral nerves staining and was therefore chosen for all subsequently performed *in-situ* hybridizations.





## B. Curriculum Vitae

### Gerald Stampfel

**Address** Dampfschiffstrasse 12/1/2  
1030 Wien, Austria

**E-Mail** gerald.stampfel@gmail.com

**Nationality** Austrian

**Date of Birth** 25.08.1983

#### Education

**2005 - 2011** Master studies in molecular biology  
University of Vienna

**2006 - 2008** Master studies in business informatics  
Vienna University of Technology

**2001 - 2006** Bachelor studies in business informatics  
Vienna University of Technology

#### Research Experience

**2009 - 2011** Intern and master student, STARK lab  
Institute of Molecular Pathology (IMP), Vienna (Austria)

**2009** Internship, FREIRE-PICOS lab  
Universidade da Coruña (Spain)

**2008** Internship, SCHNORRER lab  
Max-Planck Insititute for Biochemistry, Martinsried (Germany)

**2008** Internship, JAP lab  
Lawrence Berkeley National Laboratories (LBNL), Berkeley  
(USA)



## Bibliography

- [1] K. Adolph. A thrombospondin homologue in *Drosophila melanogaster*: cDNA and protein structure. *Gene*, 269(1-2):177–184, 2001.
- [2] W. B. L. Alkema, O. Johansson, J. Lagergren, and W. W. Wasserman. Mscan: identification of functional clusters of transcription factor binding sites. *Nucleic Acids Research*, 32(Web Server issue):W195–8, Jul 2004.
- [3] M. I. Arnone and E. H. Davidson. The hardwiring of development: organization and function of genomic regulatory systems. *Development*, 124(10):1851–64, May 1997.
- [4] S. Ashraf and Y. Ip. The Snail protein family regulates neuroblast expression of *inscuteable* and *string*, genes involved in asymmetry and cell division in *Drosophila*. *Development*, 128(23):4757, 2001.
- [5] G. Badis et al. Diversity and complexity in dna recognition by transcription factors. *Science*, 324(5935):1720–3, Jun 2009.
- [6] J. Banerji, S. Rusconi, and W. Schaffner. Expression of a beta-globin gene is enhanced by remote sv40 dna sequences. *Cell*, 27(2 Pt 1):299–308, Dec 1981.
- [7] M. Blanchette et al. Genome-wide computational prediction of transcriptional regulatory modules reveals new insights into human gene expression. *Genome Res*, 16(5):656–68, May 2006.
- [8] M. J. Blow et al. Chip-seq identification of weakly conserved heart enhancers. *Nat Genet*, 42(9):806–10, Sep 2010.
- [9] M. R. Bowl et al. An interstitial deletion-insertion involving chromosomes 2p25.3 and xq27.1, near *sox3*, causes x-linked recessive hypoparathyroidism. *J Clin Invest*, 115(10):2822–31, Oct 2005.
- [10] A. P. Boyle et al. High-resolution genome-wide in vivo footprinting of diverse transcription factors in human cells. *Genome Res*, Jan 2011.
- [11] A. H. Brand and N. Perrimon. Targeted gene expression as a means of altering cell fates and generating dominant phenotypes. *Development*, 118(2):401–15, Jun 1993.
- [12] R. A. Cameron, P. Oliveri, J. Wyllie, and E. H. Davidson. cis-regulatory activity of randomly chosen genomic fragments from the sea urchin. *Gene Expr Patterns*, 4(2):205–13, Mar 2004.

- [13] D. Choi, H. G. Goo, J. Yoo, and S. Kang. Identification of rnf2-responding loci in long-range chromatin interactions using the novel 4c-chip-cloning technology. *Journal of biotechnology*, Jan 2011.
- [14] E. P. Consortium et al. Identification and analysis of functional elements in 1genome by the encode pilot project. *Nature*, 447(7146):799–816, Jun 2007.
- [15] F. H. Crick, L. Barnett, S. Brenner, and R. J. Watts-Tobin. General nature of the genetic code for proteins. *Nature*, 192:1227–32, Dec 1961.
- [16] E. Davidson. *The regulatory genome: gene regulatory networks in development and evolution*. Academic Press, 2006.
- [17] E. H. Davidson. Network design principles from the sea urchin embryo. *Current Opinion in Genetics & Development*, 19(6):535–40, Dec 2009.
- [18] M. C. Driscoll, C. S. Dobkin, and B. P. Alter. Gamma delta beta-thalassemia due to a de novo mutation deleting the 5' beta-globin gene activation-region hypersensitive sites. *Proc Natl Acad Sci USA*, 86(19):7470–4, Oct 1989.
- [19] M. Fairchild. *Color appearance models*. Wiley, 2005.
- [20] S. Gallo, D. Gerrard, D. Miner, M. Simich, B. Des Soye, C. Bergman, and M. Halfon. REDfly v3. 0: toward a comprehensive database of transcriptional regulatory elements in Drosophila. *Nucleic Acids Research*, 39(suppl 1):D118, 2011.
- [21] S. Gallo, L. Li, Z. Hu, and M. Halfon. REDfly: a regulatory element database for Drosophila. *Bioinformatics*, 22(3):381, 2006.
- [22] K. J. Gaulton et al. A map of open chromatin in human pancreatic islets. *Nat Genet*, 42(3):255–9, Mar 2010.
- [23] M. B. Gerstein et al. Integrative analysis of the caenorhabditis elegans genome by the modencode project. *Science*, 330(6012):1775–87, Dec 2010.
- [24] P. G. Giresi and J. D. Lieb. Isolation of active regulatory elements from eukaryotic chromatin using faire (formaldehyde assisted isolation of regulatory elements). *Methods*, 48(3):233–9, Jul 2009.
- [25] F. Gong et al. The bcl2 gene is regulated by a special at-rich sequence binding protein 1-mediated long range chromosomal interaction between the promoter and the distal element located within the 3'-utr. *Nucleic Acids Research*, Feb 2011.
- [26] D. S. Gross and W. T. Garrard. Nuclease hypersensitive sites in chromatin. *Annu Rev Biochem*, 57:159–97, Jan 1988.

- [27] O. Hallikas, K. Palin, N. Sinjushina, R. Rautiainen, J. Partanen, E. Ukkonen, and J. Taipale. Genome-wide prediction of mammalian enhancers based on analysis of transcription-factor binding affinity. *Cell*, 124(1):47–59, Jan 2006.
- [28] E. E. Hare, B. K. Peterson, V. N. Iyer, R. Meier, and M. B. Eisen. Sepsid even-skipped enhancers are functionally conserved in drosophila despite lack of sequence conservation. *PLoS Genet*, 4(6):e1000106, Jun 2008.
- [29] V. Hartenstein. *Atlas of Drosophila Development*. Cold Spring Harbor Laboratory Press, 1993.
- [30] M. Hiller, T. Lin, C. Wood, and M. Fuller. Developmental regulation of transcription by a tissue-specific TAF homolog. *Genes & development*, 15(8):1021, 2001.
- [31] W. Kent, C. Sugnet, T. Furey, K. Roskin, T. Pringle, A. Zahler, et al. The human genome browser at UCSC. *Genome research*, 12(6):996, 2002.
- [32] D. Kioussis, E. Vanin, T. deLange, R. A. Flavell, and F. G. Grosveld. Beta-globin gene inactivation by dna translocation in gamma beta-thalassaemia. *Nature*, 306(5944):662–6, Jan 1983.
- [33] D.-J. Kleinjan and P. Coutinho. Cis-ruption mechanisms: disruption of cis-regulatory control as a cause of human genetic disease. *Briefings in functional genomics & proteomics*, 8(4):317–32, Jul 2009.
- [34] S. Kumar. Consortium F (2009) A knowledgebase spatiotemporal expression patterns at a genomic-scale in the fruit-fly embryogenesis (www.fly-express.net). Arizona State University, Tempe, Arizona 85287.
- [35] M. Levine and R. Tjian. Transcription regulation and animal diversity. *Nature*, 424(6945):147–51, Jul 2003.
- [36] S. MacArthur et al. Developmental roles of 21 Drosophila transcription factors are determined by quantitative differences in binding to an overlapping set of thousands of genomic regions. *Genome Biology*, 10(7):R80, 2009.
- [37] M. Markstein, P. Markstein, V. Markstein, and M. S. Levine. Genome-wide analysis of clustered dorsal binding sites identifies putative target genes in the drosophila embryo. *Proc Natl Acad Sci USA*, 99(2):763–8, Jan 2002.
- [38] M. McArthur, S. Gerum, and G. Stamatoyannopoulos. Quantification of dnasei-sensitivity by real-time pcr: quantitative analysis of dnasei-hypersensitivity of the mouse beta-globin lcr. *J Mol Biol*, 313(1):27–34, Oct 2001.
- [39] A. Miele and J. Dekker. Mapping cis- and trans- chromatin interaction networks using chromosome conformation capture (3c). *Methods Mol Biol*, 464:105–21, Jan 2009.

- [40] modENCODE Consortium et al. Identification of functional elements and regulatory circuits by drosophila modencode. *Science*, 330(6012):1787–97, Dec 2010.
- [41] I. Molina, S. Baars, J. Brill, K. Hales, M. Fuller, and P. Ripoll. A chromatin-associated kinesin-related protein required for normal mitotic chromosome segregation in *Drosophila*. *The Journal of cell biology*, 139(6):1361, 1997.
- [42] H. P. Müller-Sturm, J. M. Sogo, and W. Schaffner. An enhancer stimulates transcription in trans when attached to the promoter via a protein bridge. *Cell*, 58(4):767–77, Aug 1989.
- [43] P. L. Nagy and D. H. Price. Formaldehyde-assisted isolation of regulatory elements. *Wiley Interdiscip Rev Syst Biol Med*, 1(3):400–6, Jan 2009.
- [44] H. Ohkura, T. Torok, G. Tick, J. Hoheisel, I. Kiss, and D. Glover. Mutation of a gene for a *Drosophila* kinesin-like protein, Klp38B, leads to failure of cytokinesis. *Journal of Cell Science*, 110(8):945–954, 1997.
- [45] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11:285–296, 1975.
- [46] R.-J. T. S. Palstra. Close encounters of the 3c kind: long-range chromatin interactions and transcriptional regulation. *Briefings in functional genomics & proteomics*, 8(4):297–309, Jul 2009.
- [47] D. Papatsenko and M. Levine. Computational identification of regulatory dnas underlying animal development. *Nature Methods*, 2(7):529–34, Jul 2005.
- [48] A. Pekowska, T. Benoukraf, P. Ferrier, and S. Spicuglia. A unique h3k4me2 profile marks tissue-specific gene regulation. *Genome Res*, 20(11):1493–502, Nov 2010.
- [49] L. A. Pennacchio et al. In vivo enhancer analysis of human conserved non-coding sequences. *Nature*, 444(7118):499–502, Nov 2006.
- [50] M. Petrascheck, D. Escher, T. Mahmoudi, C. P. Verrijzer, W. Schaffner, and A. Barberis. Dna looping induced by a transcriptional enhancer in vivo. *Nucleic Acids Research*, 33(12):3743–50, Jan 2005.
- [51] B. Pfeiffer et al. Tools for neuroanatomy and neurogenetics in *Drosophila*. *Proceedings of the National Academy of Sciences*, 105(28):9715, 2008.
- [52] S. Prabhakar, F. Poulin, M. Shoukry, V. Afzal, E. M. Rubin, O. Couronne, and L. A. Pennacchio. Close sequence comparisons are sufficient to identify human cis-regulatory elements. *Genome Res*, 16(7):855–63, Jul 2006.
- [53] A. Rada-Iglesias, R. Bajpai, T. Swigut, S. A. Brugmann, R. A. Flynn, and J. Wysocka. A unique chromatin signature uncovers early developmental enhancers in humans. *Nature*, Dec 2010.

- [54] K. Robasky and M. L. Bulyk. Uniprobe, update 2011: expanded content and search tools in the online database of protein-binding microarray data on protein-dna interactions. *Nucleic Acids Research*, 39(Database issue):D124–8, Jan 2011.
- [55] R. J. Schwartz and E. N. Olson. Building the heart piece by piece: modularity of cis-elements regulating nkx2-5 transcription. *Development*, 126(19):4187–92, Oct 1999.
- [56] W. S. Simonet, N. Bucay, R. E. Pitas, S. J. Lauer, and J. M. Taylor. Multiple tissue-specific elements control the apolipoprotein e/c-i gene locus in transgenic mice. *J Biol Chem*, 266(14):8651–4, May 1991.
- [57] M. Simonis, J. Kooren, and W. de Laat. An evaluation of 3c-based methods to capture dna interactions. *Nature Methods*, 4(11):895–901, Nov 2007.
- [58] R. Sokal, C. Michener, and U. of Kansas. A statistical method for evaluating systematic relationships. 1958.
- [59] A. Stark et al. Discovery of functional elements in 12 drosophila genomes using evolutionary signatures. *Nature*, 450(7167):219–32, Nov 2007.
- [60] Steve Eddins. Re: About the matlab function of "roifill", Apr. 2009. <http://sci.tech-archive.net/Archive/sci.image.processing/2009-04/msg00045.html>.
- [61] A. I. Su et al. Large-scale analysis of the human and mouse transcriptomes. *Proc Natl Acad Sci USA*, 99(7):4465–70, Apr 2002.
- [62] A. I. Su et al. A gene atlas of the mouse and human protein-encoding transcriptomes. *Proc Natl Acad Sci USA*, 101(16):6062–7, Apr 2004.
- [63] M. Tanaka-Matakatsu, T. Uemura, H. Oda, M. Takeichi, and S. Hayashi. Cadherin-mediated cell adhesion and cell motility in *Drosophila* trachea regulated by the transcription factor Escargot. *DEVELOPMENT-CAMBRIDGE-*, 122:3697–3705, 1996.
- [64] P. Tomancak et al. Global analysis of patterns of gene expression during *Drosophila* embryogenesis. *Genome biology*, 8(7):R145, 2007.
- [65] V. Viprakasit, C. L. Harteveld, H. Ayyub, J. S. Stanley, P. C. Giordano, W. G. Wood, and D. R. Higgs. A novel deletion causing alpha thalassemia clarifies the importance of the major human alpha globin regulatory element. *Blood*, 107(9):3811–2, May 2006.
- [66] A. Visel, J. A. Akiyama, M. Shoukry, V. Afzal, E. M. Rubin, and L. A. Pennacchio. Functional autonomy of distant-acting human enhancers. *Genomics*, 93(6):509–13, Jun 2009.
- [67] A. Visel et al. Chip-seq accurately predicts tissue-specific activity of enhancers. *Nature*, 457(7231):854–8, Feb 2009.

- [68] A. Visel, S. Minovitsky, I. Dubchak, and L. A. Pennacchio. Vista enhancer browser—a database of tissue-specific human enhancers. *Nucleic Acids Research*, 35(Database issue):D88–92, Jan 2007.
- [69] A. Woolfe et al. Highly conserved non-coding sequences are associated with vertebrate development. *Plos Biol*, 3(1):e7, Jan 2005.
- [70] X. yong Li et al. Transcription factors bind thousands of active and inactive regions in the drosophila blastoderm. *Plos Biol*, 6(2):e27, Feb 2008.
- [71] C. Zhu et al. High-resolution dna-binding specificity analysis of yeast transcription factors. *Genome Res*, 19(4):556–66, Apr 2009.
- [72] R. P. Zinzen, C. Girardot, J. Gagneur, M. Braun, and E. E. M. Furlong. Combinatorial binding predicts spatio-temporal cis-regulatory activity. *Nature*, 462(7269):65–70, Nov 2009.