Binary pattern dictionary learning for gene expression representation in drosophila imaginal discs

Jiří Borovec, Jan Kybic

Center for Machine Perception,
Department of Cybernetics, Faculty of Electrical Engineering,
Czech Technical University in Prague, Czech Republic

Abstract. We present an image processing pipeline which accepts a large number of images, containing spatial expression information for thousands of genes in Drosophila imaginal discs. We assume that the gene activations are binary and can be expressed as a union of a small set of non-overlapping spatial patterns, yielding a compact representation of the spatial activation of each gene. This lends itself well to further automatic analysis, with the hope of discovering new biological relationships. Traditionally, the images were labeled manually, which was very time consuming. The key part of our work is a binary pattern dictionary learning algorithm, that takes a set of binary images and determines a set of patterns, which can be used to represent the input images with a small error. We also describe the preprocessing phase, where input images are segmented to recover the activation images and spatially aligned to a common reference. We compare binary pattern dictionary learning to existing alternative methods on synthetic data and also show results of the algorithm on real microscopy images of the Drosophila imaginal discs.

1 Introduction

The fruit fly Drosophila is a frequently used valuable subject in modern experimental biology due to their short life cycle and genetic similarity to humans [1]. Large scale mapping of the gene expressions was performed in embryos [2,3] as well as in imaginal discs [4,5], which are essential for the initial development of the adult fly. The expressed gene is highlighted using molecular biology methods and microscopy images of many thousands samples are acquired.

The final goal is to understand the role of the different genes by comparing locations, where the genes are expressed, with the known information about the function of the different areas. To reduce the dimensionality of the problem and enable an efficient statistical analysis, the observed spatial expressions are described by a set of labels from a limited, application specific dictionary, called an 'atlas'. Example labels for the leg imaginal disc are 'dorsal' or 'ventral' but also 'stripes' or 'ubiquitous'. Given such sets of labels, correlations with gene ontologies can be then found using data mining methods [6,7,8].

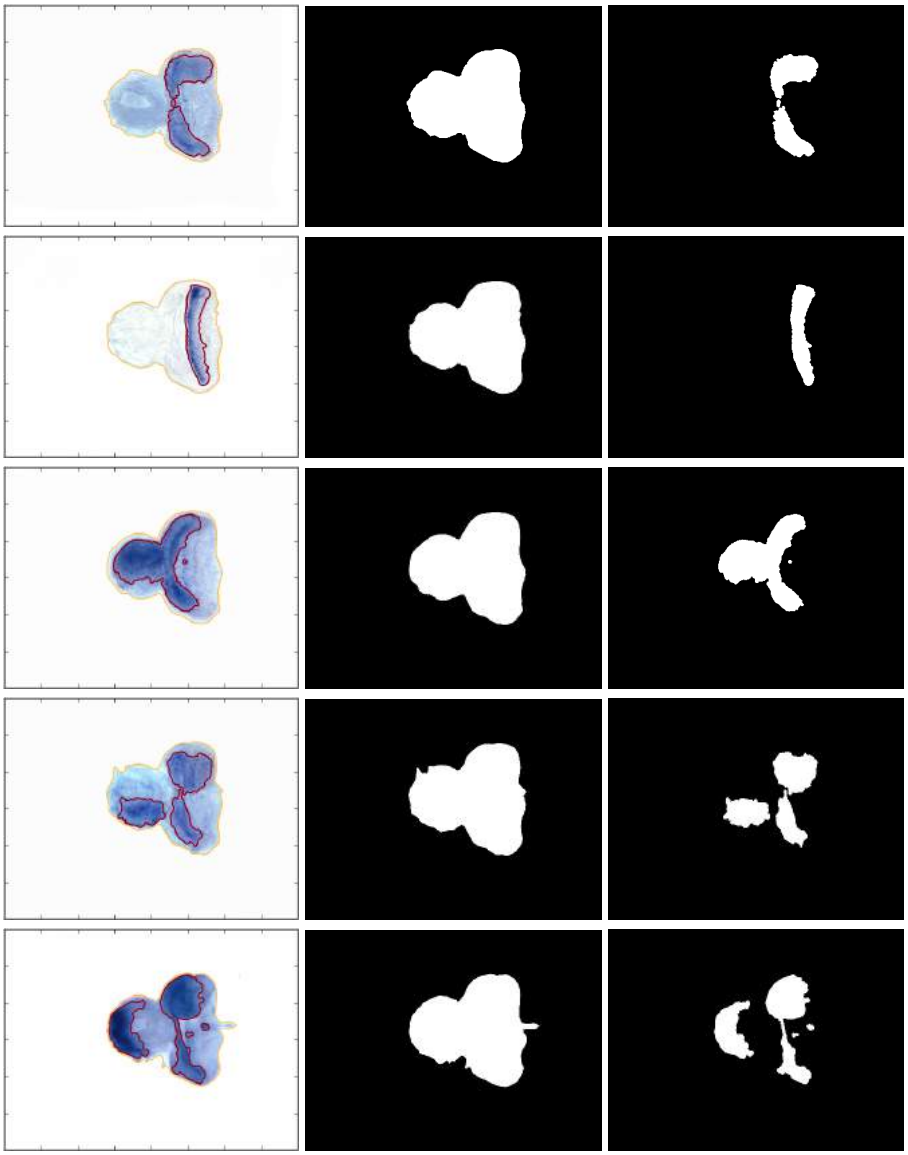


Fig. 1. Presenting samples of aligned Drosophila imaginal discs (eye antenna discs type) and their segmentation. First, we show the sensed images (a) with marked contour of segmented disc (orange) and gene expression (red) followed by visualisation of the segmented discs (b) and segmented gene expressions (c).

The dictionary as well as the labels for individual images are usually determined manually or semi-manually [2,9,10], which is extremely intensive work. Some automatic methods exists, based on e.g. sparse Bayesian factor models [11] or non-negative matrix factorization [12]. In contrast to these methods, we assume in this work that the activation and the patterns are inherently binary. We further assume that the patterns, corresponding to anatomically defined zones, are compact and non-overlapping. These constraints should increase the robustness of the estimation and yield more biologically plausible results.

1.1 State-of-the-art

Let a matrix $X \in \mathbb{R}^{|\Omega| \times N}$ be a rearranged set of pixels of N images with pixel coordinates Ω. In our case, the images are assumed to be aligned and the pixel intensities to correspond to the gene activations. A linear decomposition of X can be found by minimizing

$$\min_{Y,W} \|X - Y \cdot W\|^2 \tag{1}$$

where $Y \in \mathbb{R}^{|\Omega| \times L}$ corresponds to a dictionary (or 'atlas') with L patterns and $W \in \mathbb{R}^{L \times N}$ are image specific weights. We shall give a few examples of known methods, differing in additional assumptions and constraints. Built on the well-known PCA, sparse Principal Component Analysis [13] (sPCA) assumes the weights W to be sparse. Fast Independent Component Analysis [14] (FastICA) seeks for spatial independence of the patterns. Dictionary Learning [15] (DL) with Matching Pursuit is a greedy iterative approximation method with many variants, mainly in the field of sparse linear approximation of signals. Non-negative Matrix Factorization [16] (NMF) adds the non-negativity constraints, while sparse Bayesian models add a probabilistic prior on the weights, encouraging sparsity. Both methods were used for estimating gene expression patterns in Drosophila embryos [11,12] (see Berkeley Drosophila Genome Project[1]).

There is far less literature in the case of binary X, Y, or W. If the requirement of spatial compactness of the patterns is dropped, then the problem is called binary matrix factorization [17,18] and is often used in data mining. Simplifying further to allow only one pattern per image leads to the problem of vector quantization [19].

2 Method

We shall now describe the complete pipeline consists of preprocessing (segmentation and registration) and atlas estimation via Binary Pattern Dictionary Learning (BPDL), as illustrated in Fig. 2.

[1] `http://insitu.fruitfly.org/cgi-bin/ex/insitu.pl`

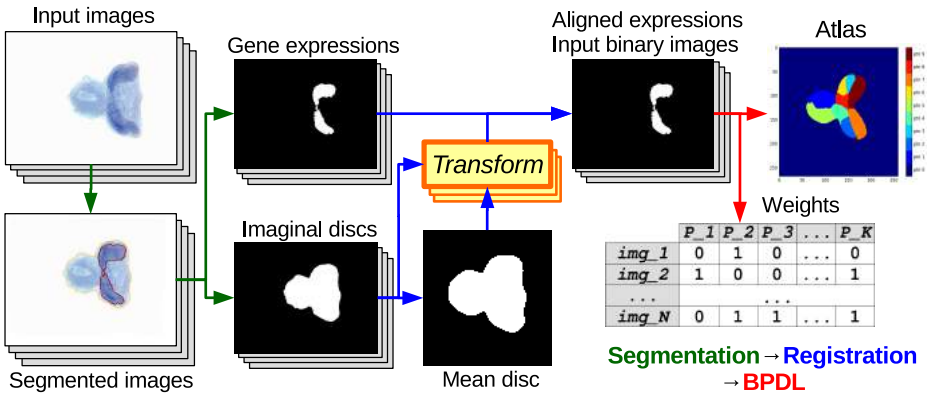


Fig. 2. A flowchart of the complete pipeline for processing images of Drosophila imaginal discs: (i) a pixel-wise segmentation into 3 classes (background, imaginal disk and gene expression); (ii) registration of binary segmented imaginal discs onto a reference shape (disc prototype); (iii) atlas estimation from aligned binary gene expressions.

2.1 Preprocessing

Given a set of images of imaginal discs (see Fig. 1a) containing both anatomical and gene expression information, preprocessing is applied to obtain a set of segmented and aligned gene expression images, which serves as input for the subsequent binary dictionary learning (see Fig. 2).

Segmentation. We first segment the input images into three clases (background, imaginal disc, gene activation) by the following steps [20]:

a) calculate SLIC superpixels [21];
b) calculate superpixel colour features — mean, median, and variance;
c) estimating a Gaussian Mixture Model (GMM) with one component per class,
d) calculate superpixel-wise class probabilities based on the GMM;
e) apply Graph Cut [22] to estimate a spatially regularized segmentation;
f) post-process, e.g. suppress very small regions and identifying the imaginal disc component.

Registration. For each of the four disc types, a reference shape is calculated as the mean disc shape over all images. Then all other images are registered to the reference shape. We use a fast elastic registration algorithm [23], which works directly on the segmented images, transforming the disc shapes by aligning their contours, ignoring the activations (see Fig. 1b). The activations (Fig. 1c) are then aligned using the recovered transformation.

Algorithm 1 General schema of BPDL algorithm.

1: initialise atlas $\mathbf{y}$
2: **while** not converged **do**
3: update weights $\mathbf{w} \in \mathbf{W}$
4: reinitialise empty patterns in $\mathbf{y}^*$
5: update atlas $\mathbf{y}^*$ via Graph Cut
6: **end while**

2.2 Binary pattern dictionary learning — problem definition

Let us define the image pixels as $\Omega \subseteq \mathbb{Z}^d$, with $d = 2$, and the input binary image as $\mathbf{g} : \Omega \to \{0, 1\}$. Our task is to find an atlas $\mathbf{y} : \Omega \to \mathbb{L}$, with labels $\mathbb{L} = [0, \ldots, K]$, assigning to each pixel either a background (label $l = 0$), or one of the labels (patterns, set of equal labels) $1, \ldots, K$. Each binary weight vector $\mathbf{w} : \mathbb{L} \to \{0, 1\}$ yields an image $\hat{\mathbf{g}}$ as a union of the selected patterns in atlas $\mathbf{y}$

$$\hat{\mathbf{g}} = \sum_{l \in \mathbb{L}} \mathbf{w}_l \cdot [\![\mathbf{y} = l]\!] \tag{2}$$

where $[\![\cdot]\!]$ denotes the Iverson bracket. Note that this is a special case of (1), with all variables binary. Note also, that in this representation, the patterns cannot overlap. The approximation error on one image $\mathbf{g}$ and its representation by $\mathbf{y}$ and $\mathbf{w}$ is the Hamming distance

$$F(\mathbf{g}, \mathbf{y}, \mathbf{w}) = \sum_{i \in \Omega} [\![\mathbf{g}_i \neq \hat{\mathbf{g}}_i]\!] = \sum_{i \in \Omega} \left| \mathbf{g}_i - \sum_{l \in \mathbb{L}} \mathbf{w}_l \cdot [\![\mathbf{y} = l]\!] \right| \tag{3}$$

To encourage spatial compactness of the estimated atlas, we shall penalize differences between neighboring pixels i, j in the atlas

$$H(\mathbf{y}) = \sum_{\substack{i,j \in \Omega,\, i \neq j, \\ d(i,j)=1}} [\![\mathbf{y}_i \neq \mathbf{y}_j]\!] \tag{4}$$

where the Kronecker delta $[\![\mathbf{y}_i \neq \mathbf{y}_j]\!]$ for all combinations of $\mathbf{y}_{i,j} \in \mathbb{L}$ can be represented as a square matrix with zeros on the main diagonal and ones otherwise

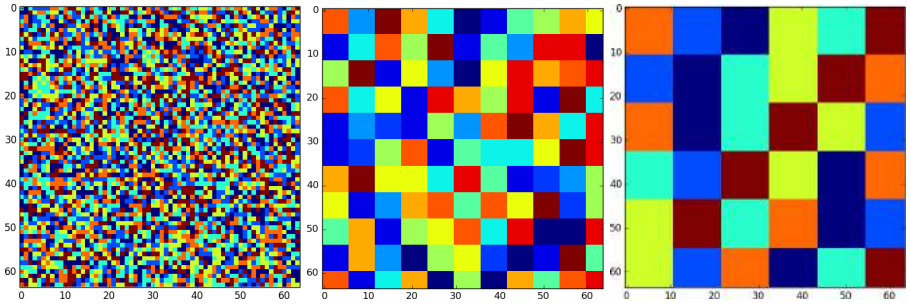
The optimal atlas and the associated weights are found by optimizing the mean approximation error for all N images

$$\mathbf{y}^*, \mathbf{w}^* = \arg\min_{\mathbf{y}, \mathbf{W}} \frac{1}{N} \sum_n F(\mathbf{g}^n, \mathbf{y}, \mathbf{w}^n) \;\; + \beta \cdot H(\mathbf{y}) \tag{5}$$

where the matrix $\mathbf{W}$ contains all weights $\mathbf{w}^n$ for $n \in [0, \ldots, N]$, and β is the spatial regularization coefficient. Sufficiently large β force the labelling to have all patterns to be connected.

2.3 BPDL — Alternating minimization

The criterion (5) is minimized alternately with respect to atlas $\mathbf{y}$ and weights $\mathbf{w} \in \mathbf{W}$, (see Algorithm 1).



(a) (b) (c)

Fig. 3. Random atlas initialization with patch sizes 1 pixel (a), $m/2K$ pixels (b), and m/K pixels (c), for $K = 6$ and $m = 64$ pixels being the image size.

Initialization. We initialize the atlas with randomly labeled patches on a regular grid, with user-defined sizes; see Fig. 3 for examples.

Update weights $\mathbf{W}$. With the atlas $\mathbf{y}$ fixed, we estimate the weights $\mathbf{w}^n$ for each image $\mathbf{g}^n$ independently. It turns out that $F(\mathbf{g}^n, \mathbf{y}, \mathbf{w})$ is minimized with respect to w_l, if the majority of pixels in the pattern $l \in \mathbf{y}$ agree with the image. We set

$$w_l = [\![P(\mathbf{g}, \mathbf{y}, l) \geq \sigma]\!] \qquad \text{where } \sigma = 1 \tag{6}$$

$$\text{and} \quad P(\mathbf{g}, \mathbf{y}, l) = \frac{\sum_{i \in \Omega, \mathbf{y}_i = l} [\![\mathbf{g}_i = 1]\!]}{\sum_{i \in \Omega, \mathbf{y}_i = l} [\![\mathbf{g}_i \neq 1]\!]} = \frac{\left| [\![\mathbf{y} = l]\!] \right|}{\sum_{i \in \Omega, \mathbf{y}_i = l} (1 - \mathbf{g}_i)} - 1 \tag{7}$$

We temporarily reduce σ in the initial stage of the algorithm, otherwise very few patterns might be selected.

Reinitialize empty patterns. During the weight calculation step, some pattern may not have been used for any image. This is wasteful, unless the reconstruction is already perfect, we can always improve it by adding another pattern. We iterate the following procedure until all K labels are used:

1. find an image $\mathbf{g}^n$ with the largest unexplained residual $[\![\mathbf{g}^n \wedge \neg \hat{\mathbf{g}}^n]\!]$
2. find the largest connected component c of this residual and assign label $l \notin \mathbf{y}$;
3. calculate weights w_l^n for the new label l for all images $\mathbf{g}^n \in \mathbf{G}$ using (6).

Update of atlas $\mathbf{y}$. With the weight vectors $\mathbf{W}$ fixed, finding the atlas $\mathbf{y}$ is a discrete labeling problem. We can rewrite the criterion in (5) as

$$\frac{1}{N} \sum_{i \in \Omega} \underbrace{\sum_{n} \left| \mathbf{g}_i^s - \sum_{l \in \mathbb{L}} \mathbf{w}_l^s \cdot [\![\mathbf{y} = l]\!] \right|}_{U_i(y_i)} + \sum_{\substack{i,j \in \Omega,\, i \neq j, \\ d(i,j)=1}} [\![\mathbf{y}_i \neq \mathbf{y}_j]\!] \tag{8}$$

which can be solved for example with Graph Cut [22] and alpha expansion.

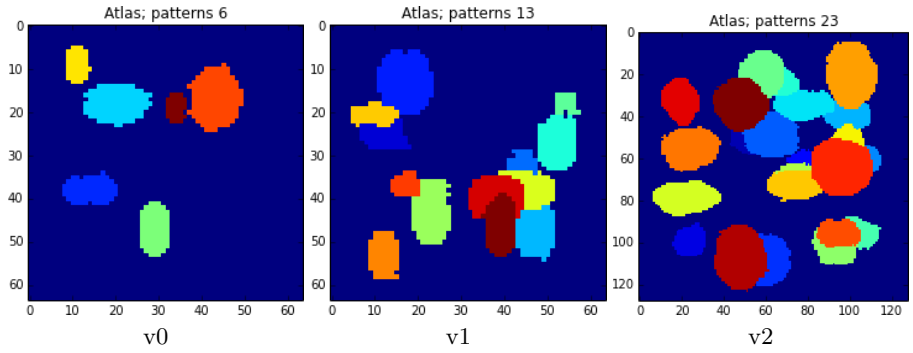


Fig. 4. Visualisation of the generated atlases for the three created synthetic datasets containing $K = 6$ (v0), $K = 13$ (v1) and $K = 23$ (v2) patterns, with image sizes 64×64 (v0, v1) and 128×128 (v2) pixels.

3 Experiments

We evaluate the performance (atlas similarity and descriptiveness and elapsed time) of the algorithm on both synthetic and real images.

3.1 Alternative methods

We have compared our BPDL with the following methods: NMF [16], FastICA [14], SparsePCA [13] and Dictionary Learning [15] (DL). All methods were implemented in the scikit-learn[2] library.

Binarization of continues components. To obtain a binary atlas $\mathbf{y}$ from a continuous matrix $Y \in \mathbb{R}^{|\Omega| \times L}$, we select the component with a maximal value in each pixel position $i \in \Omega$, i.e.

$$\mathbf{y}_i = \arg\max_{l \in \mathbb{L}} Y_i^l$$

3.2 Synthetic dataset

We generated three synthetic datasets (v0, v1, v2) representing already segmented and aligned binary images based on a random atlas and random pattern weights. The patterns are deformed ellipses. The datasets differ in image size and the true number of patterns in atlas (see Fig. 4).

Each dataset is further divided into three sub-sets, each containing 1200 input images (Fig. 5):

1. **pure:** images generated from equation (2)

[2] `http://scikit-learn.org/stable/`

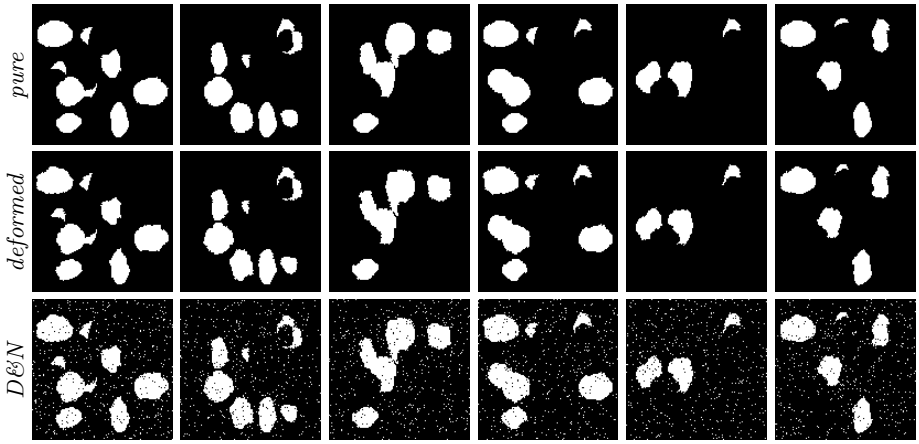


Fig. 5. We show sample images from the synthetic dataset v2. The tree rows represents the 3 sub-sets of input images: pure, deformed and deformed with random binary noise (denoted D&N).

2. **deform:** pure images (1) independently transformed by a small elastic B-spline deformation with the maximum amplitude of $0.2m$, with $m = \sqrt{|\Omega|}$.
3. **deform & noise (D&N):** deformed images (2) with random binary noise (randomly flipping 10% pixels).

3.3 Evaluation metrics

Atlas comparison The difference between atlases is given by the Adjusted Rand Score[3] (ARS), which gives similarity value in the range $(0, 1)$, with 1 being a perfect match.

Reconstruction difference With the estimated atlas $\mathbf{y}$ and pattern weights $\mathbf{w}^n \in \mathbf{W}$ for each particular image $\mathbf{g}^n \in \mathbf{G}$ we reconstruct each input image $\hat{\mathbf{g}}^n$ (see eq. (2), Fig. 6), the approximation error is averaged over all images:

$$R(\mathbf{G}, \mathbf{y}, \mathbf{W}) = \frac{1}{N \cdot |\Omega|} \sum_{n} F(\mathbf{g}^n, \mathbf{y}, \mathbf{w}^n) = \frac{1}{N \cdot |\Omega|} \sum_{n} \sum_{i} |\mathbf{g}_i^n - \hat{\mathbf{g}}_i^n| \quad (9)$$

In case of the synthetic datasets we always compare the reconstructed images to the pure input images.

3.4 Comparison on synthetic datasets

In Table (1) we show the accuracy of reconstructing the atlas (measured by ARS), the mean approximation error R, and elapsed time for all datasets and their modifications. The number of patterns was set to the true value K.

[3] `https://en.wikipedia.org/wiki/Rand_index`

datasets		**NMF** [16]	**FastICA** [14]	**sPCA** [13]	**DL** [15]	**BPDL** *
v0		*(size* 64×64 *px, 6 patterns)*				
pure	ARS	**1.0**	**1.0**	**0.961**	**1.0**	**0.999**
	diff.	**0.0**	**0.0**	**0.002**	**0.0**	**0.0**
	time	**2.780**	168.476	30.842	304.51	**6.658**
deform	ARS	0.775	**0.921**	0.769	0.777	**0.993**
	diff.	0.014	**0.004**	0.0213	0.014	**0.0**
	time [s]	**1.697**	141.527	22.833	279.87	**4.766**
D&N	ARS	0.048	**0.778**	0.002	0.066	**0.999**
	diff.	0.033	**0.014**	0.033	0.033	**0.0**
	time [s]	**2.005**	229.47	24.907	598.83	**6.774**
v1		*(size* 64×64 *px, 13 patterns)*				
pure	ARS	**1.0**	**1.0**	0.992	**0.995**	**0.999**
	diff.	**0.0**	**0.0**	0.0298	**0.019**	**0.0**
	time	**2.333**	340.32	18.291	737.47	**6.029**
deform	ARS	0.785	**0.948**	0.780	0.779	**0.992**
	diff.	0.017	**0.004**	0.029	0.033	**0.005**
	time [s]	**4.001**	312.18	15.000	700.03	**7.561**
D&N	ARS	0.091	**0.878**	0.009	0.0727	**0.951**
	diff.	0.048	**0.010**	0.061	0.0499	**0.003**
	time [s]	**4.490**	439.04	11.420	697.599	**9.562**
v2		*(size* 128×128 *px, 23 patterns)*				
pure	ARS	**1.0**	**1.0**	**0.989**	**1.0**	**0.999**
	diff.	**0.0**	**0.0**	0.037	**0.0**	**0.005**
	time [s]	**82.329**	5533.4	460.82	14786.	**88.260**
deform	ARS	0.818	**0.846**	0.801	0.807	**0.970**
	diff.	0.019	**0.015**	0.056	0.046	**0.004**
	time [s]	**144.10**	5683.2	477.47	13619.	**165.22**
D&N	ARS	0.120	**0.612**	0.024	0.144	**0.877**
	diff.	**0.036**	**0.036**	0.092	0.039	**0.013**
	time [s]	**77.399**	6912.9	485.44	13729.	**289.51**

Table 1. Performance comparison on the synthetic datasets. We show the atlas ARS (Adjusted Rand Score), approximation error R (9), and processing time in seconds. We colour the best (blue) and the second best (cyan) result. All experiments were performed on the same computer, in a single thread configuration. The results shows that all methods work well on the 'pure' sub-set. For the deformed and also noise images the best results was obtained by BPDL. The fastest method was NMF, followed by BPDL.

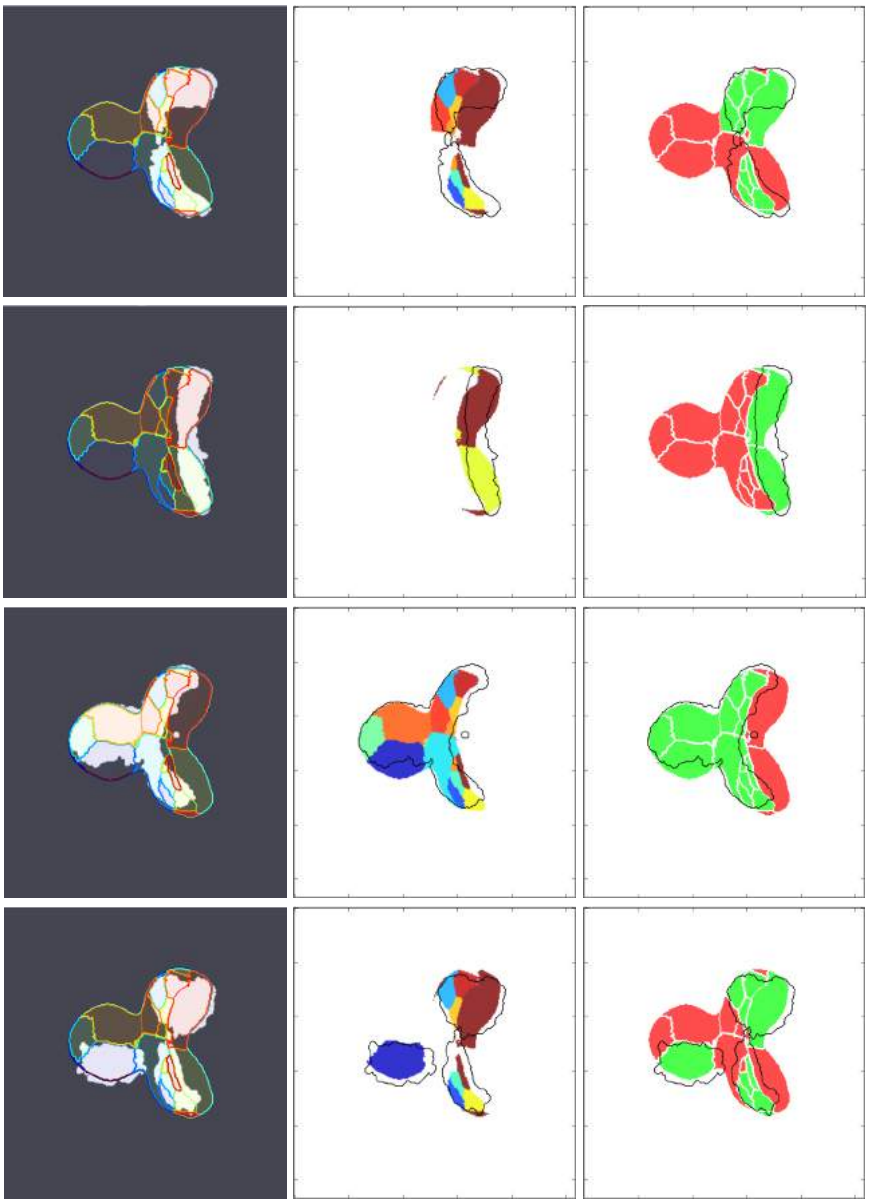


Fig. 6. Visualization of the reconstruction of real images (already showen in Fig. 1) in three different ways: (a) The input binary segmentation of gene expression overlapped by the atlas pattern contours. (b) The individual atlas patterns (in color) with the binary input gene expression segmentation overlaid (black contour). (c) Used (green) versus unused (red) atlas patterns with contour of the segmented input expression boundary (black).

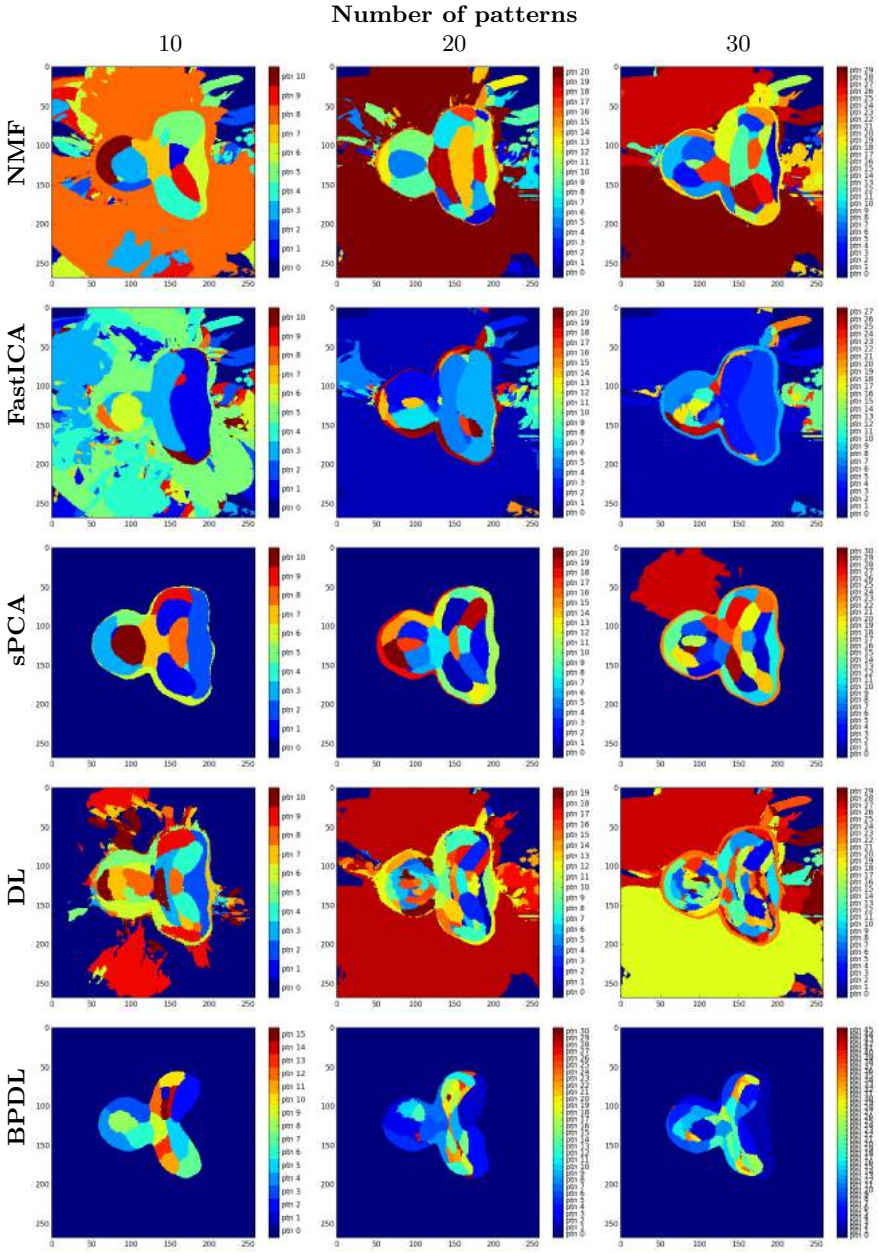


Fig. 7. Presenting estimated atlases by all methods with different number of estimated patterns K.

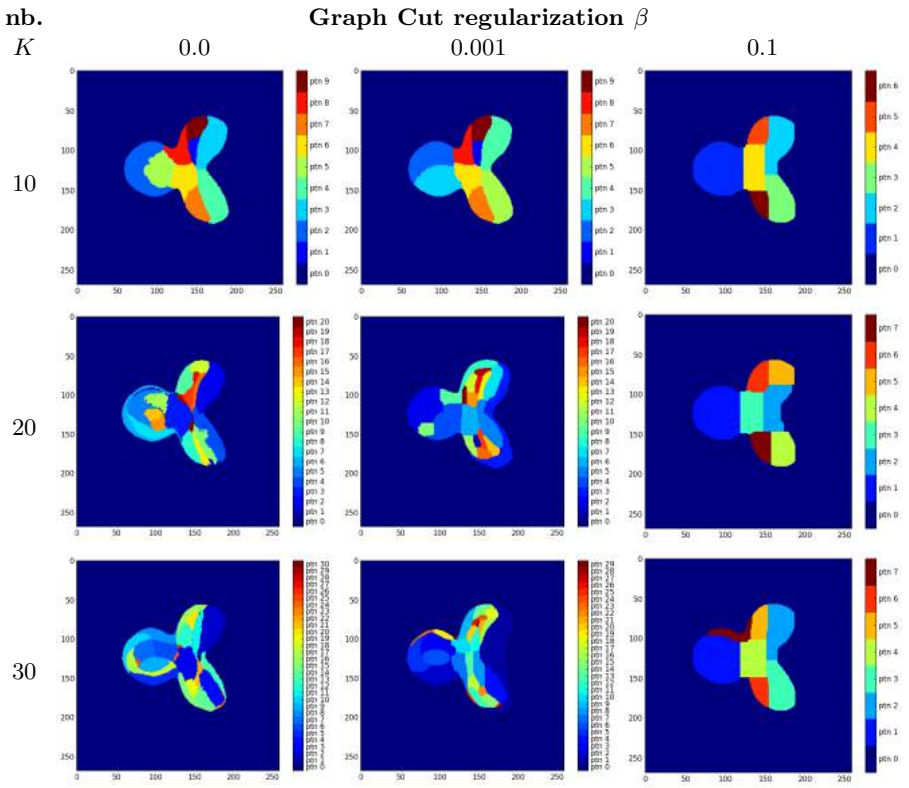


Fig. 8. Visualization of the estimated atlas for Drosophila images (eye type imaginal discs) as a function of the number of estimated patterns K and the Graph Cut regularization parameter β.

We can say that on the 'pure' images, all methods work well. In other cases, the accuracy of our method (as measured by ARS and R) is better. The fastest method is the NMF (on average twice faster than BPDL) but its results are poor. On the other hand FastICA gives the second best quality results after BPDL but is much slower (on average 40 times slower then BPDL).

3.5 Comparison on real images

We applied all methods on segmented gene expressions images of the Drosophila imaginal discs varying the number of patterns $K \in \{10, 20, 30\}$. Several reconstruction examples for BFDL are shown in Fig. 6. Looking at the estimated atlases (Fig. 7) we found that NMF, FastICA and DL have difficulty to identify background and often produce very small regions. Example atlases by BFDL on all four considered disc types are shown in Fig. 9.

Method	**Number of patterns** K			**Time [min]**
	10	20	30	
NMF [16]	0.0939	0.0823	0.0723	**10**
FastICA [14]	0.1197	0.0779	0.0485	24
sPCA [13]	**0.0476**	**0.0413**	**0.0352**	477
DL [15]	0.0939	0.0648	0.0596	338
BPDL *	**0.0467**	**0.0395**	**0.0361**	**20**

Table 2. Reconstruction difference R on real images of imaginal disc (eye type) by all tested methods for three different assumed numbers of patterns K.

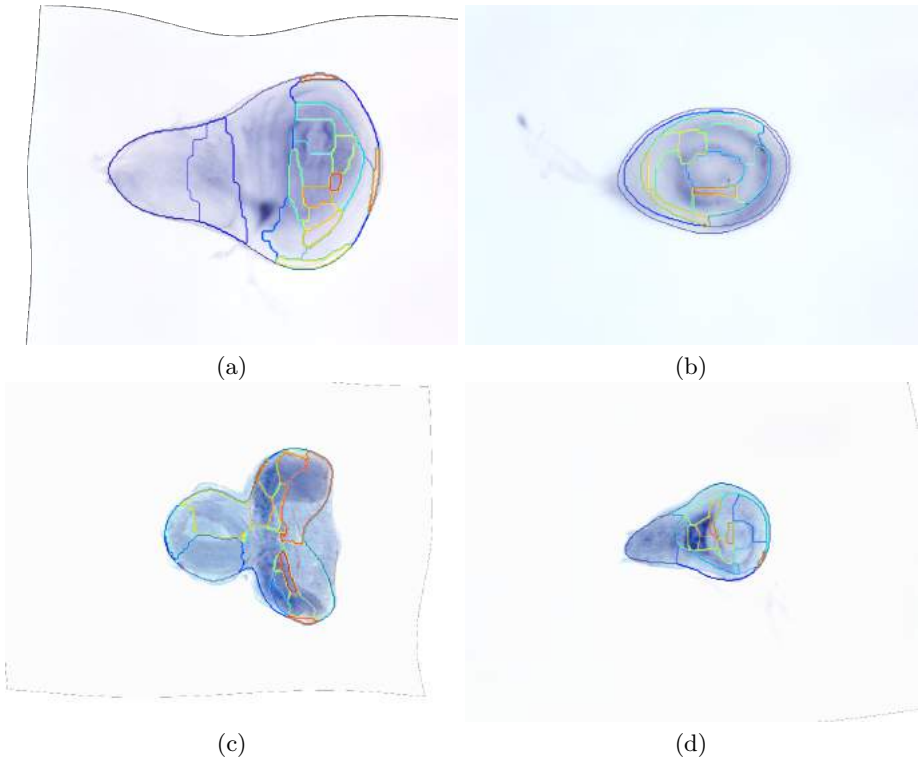


Fig. 9. Sample images of each imaginal disc types: wing (a), leg (b), eye (c), haltere (d) with the atlases estimated by BPDL shown as contour overlays for number of patterns $K = 20$.

The effect of the Graph Cut regularization parameter β is shown in Fig. 8. A value of $\beta = 0.001$ was found to perform best by subjective evaluation and it was used in all other experiments.

4 Conclusion

This paper addresses automatic image analysis of Drosophila imaginal discs, focusing on the problem of finding an atlas of atomic gene expression from the images. Unlike alternative methods, we assume that the atlas and its coefficients are binary and our proposed method (BPDL) estimates an atlas of binary patterns directly by an iterative procedure.

On synthetic datasets, BPDL achieves the best overall quality results, with a very reasonable computational complexity. On real datasets, BPDL produces similar quality atlas and reconstruction as the SparsePCA method, while being much faster.

The extracted image labels will be further processed by data mining methods. The proposed binary pattern dictionary learning can be applied any time a large set of binary images should be represented by a small dictionary.

Acknowledgement. This work was supported by the Czech Science Foundation project 14-21421S and by the Grant Agency of the Czech Technical University in Prague under the grant SGS15/154/OHK3/2T/13.

References

1. Medzhitov R., Preston-Hurlburt P., Janeway C.A. Jr: A human homologue of the Drosophila Toll protein signals activation of adaptive immunity. Nature **388** (1997) 394–397
2. Tomancak, P., Berman, B.P., Beaton, A., Weiszmann, R., Kwan, E., Hartenstein, V., Celniker, S.E., Rubin, G.M.: Global analysis of patterns of gene expression during Drosophila embryogenesis. Genome biology **8** (2007) R145
3. Hammonds, A.A.S., Bristow, C.C.a., Fisher, W.W., Weiszmann, R., Wu, S., Hartenstein, V., Kellis, M., Yu, B., Frise, E., Celniker, S.E.: Spatial expression of transcription factors in Drosophila embryonic organ development. Genome biology **14** (2013) R140
4. Brower, D.L.: Engrailed gene expression in Drosophila imaginal discs. The EMBO journal **5** (1986) 2649–2656
5. Ahammad, P., Harmon, C.L., Hammonds, A., Sastry, S.S., Rubin, G.M.: Joint Nonparametric Alignment for Analizing Spatial Gene Expression Patterns in Drosophila Imaginal Discs. Proc. CVPR (2005)
6. Jiang, D., Tang, C., Zhang, A.: Cluster analysis for gene expression data: A survey. IEEE Transactions on Knowledge and Data Engineering **16** (2004) 1370–1386
7. Kim, J., Kim, K., Kim, J.H.: Semantic Signature: Comparative Interpretation of Gene Expression on a Semantic Space. In: Computational and Mathematical Methods in Medicine. Volume 2016. (2016) 1–10
8. Klema, J., Malinka, F., Zelezny, F.: Semantic biclustering: a new way to analyze and interpret gene expression data. In: Bioinformatics Research and Applications, Minsk, Belarus, Springer (2016) 332–3
9. Tweedie, S., Ashburner, M., Falls, K., Leyland, P., McQuilton, P., et al.: FlyBase: Enhancing Drosophila Gene Ontology annotations. Nucleic Acids Research **37** (2009) 555–559

10. Tomancak, P., Beaton, A., Weiszmann, R., Kwan, E., Shu, S., Lewis, S.E., Richards, S., Ashburner, M., Hartenstein, V., Celniker, S.E., Rubin, G.M.: Systematic determination of patterns of gene expression during Drosophila embryogenesis. Genome biology **3** (2002) RESEARCH0088
11. Pruteanu-Malinici, I., Mace, D.L., Ohler, U.: Automatic annotation of spatial expression patterns via sparse bayesian factor models. PLoS Computational Biology **7** (2011)
12. Wu, S., Joseph, A., Hammonds, A.S., Celniker, S.E., Yu, B., Frise, E.: Stability-driven nonnegative matrix factorization to interpret spatial gene expression and build local gene networks. Proceedings of the National Academy of Sciences **113** (2016) 201521171
13. Zou, H., Hastie, T., Tibshirani, R., Johnstone, I., Lu, A.: Sparse principal component analysis. Journal of Computational and Graphical Statistics **15** (2006) 1–29
14. Hyvarinen, A.: Fast and Robust Fixed-Point Algorithm for Independent Component Analysis. IEEE Trans. Neur. Net. **10** (1999) 626–634
15. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09. (2009) 1–8
16. Lin, C.J.: Projected gradient methods for nonnegative matrix factorization. Neural computation **19** (2007) 2756–2779
17. Belohlavek, R., Vychodil, V.: Discovery of optimal factors in binary data via a novel method of matrix decomposition. Journal of Computer and System Sciences **76** (2010) 3–20
18. Zhang, Z.Y., Li, T., Ding, C., Ren, X.W., Zhang, X.S.: Binary matrix factorization for analyzing gene expression data. Data Mining and Knowledge Discovery **20** (2010) 28–52
19. Gersho, A., Gray, R.M.: Vector Quantization and Signal Compression. Kluwer Academic Press **159** (1992) 760
20. Borovec, J.: Fully automatic segmentation of stained histological cuts. In Husník, L., ed.: 17th International Student Conference on Electrical Engineering, Prague, CTU in Prague (2013) 1–7
21. Achanta, R., Shaji, A.: SLIC Superpixels Compared to State-of-the-art Superpixel Methods. Pattern Analysis and Machine Intelligence, IEEE **34** (2012) 2274 – 2282
22. Boykov, Y., Veksler, O.: Fast approximate energy minimization via graph cuts. Pattern Analysis and Machine Intelligence, IEEE **23** (2001) 1222–1239
23. Kybic, J., Dolejsi, M., Borovec, J.: Fast registration of segmented images by normal sampling. In: Bio Image Computing (BIC) workshop at CVPR. (2015) 11–19