Image segmentation & BPDL in microscopy

Jiri Borovec
@ CMP, FEE, CTU in Prague

2018

http://cmp.felk.cvut.cz/~borovji3/

https://www.researchgate.net/publication/323120618

Resources

- Publications:
  - Borovec J. (2017). **Automatic analysis of gene expressions in Drosophila microscopy images.** ftp://cmp.felk.cvut.cz/pub/cmp/articles/borovec/Thesis-TR-2017-07.pdf
  - Borovec J., Svihlik J., Kybic J., Habart D. (2017). **Supervised and unsupervised segmentation using superpixels, model estimation, and Graph Cut.** SPIE Journal of Electronic Imaging 26(6), 061610, http://doi.org/10.1117/1.JEI.26.6.061610
  - Borovec J., Kybic J., Nava R. (2017) **Detection and Localization of Drosophila Egg Chambers in Microscopy Images.** In: Wang Q., Shi Y., Suk HI., Suzuki K. (eds) Machine Learning in Medical Imaging. MLMI 2017. LNCS, vol 10541. Springer, Cham. http://doi.org/10.1007/978-3-319-67389-9_3
  - Borovec J., Kybic J., Sugimoto, A. (2017). **Region growing using superpixels with learned shape prior.** SPIE Journal of Electronic Imaging 26(6), 061611, http://doi.org/10.1117/1.JEI.26.6.061611
  - Borovec J., Kybic J. (2016) **Binary Pattern Dictionary Learning for Gene Expression Representation in Drosophila Imaginal Discs**. In: Computer Vision – ACCV 2016 Workshops. Lecture Notes in Computer Science, vol 10117, Springer, http://doi.org/10.1007/978-3-319-54427-4_40

Overview

1. Problem statement
2. Used datasets
3. Instance segmentation
   a. Structure (tissue) segmentation on superpixels
   b. Center detection and ellipse fitting
   c. Region growing on superpixels
4. Binary Pattern Dictionary Learning
5. Conclusion & Future work

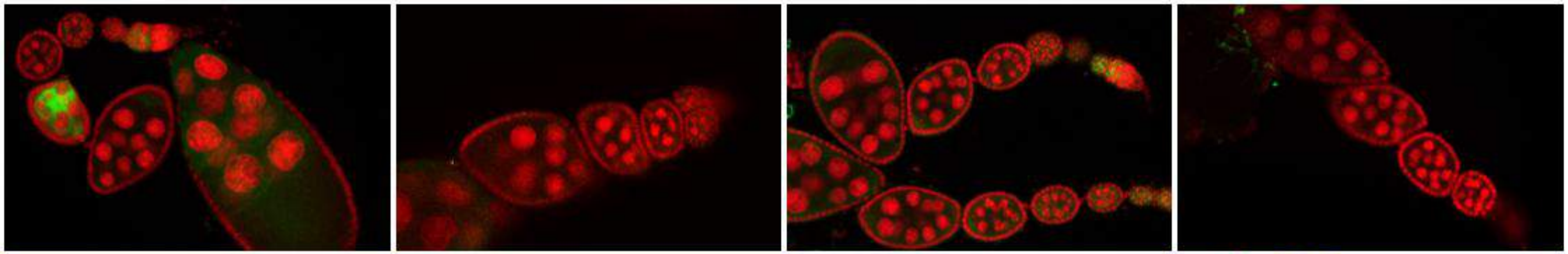
Motivation

Why Drosophila?

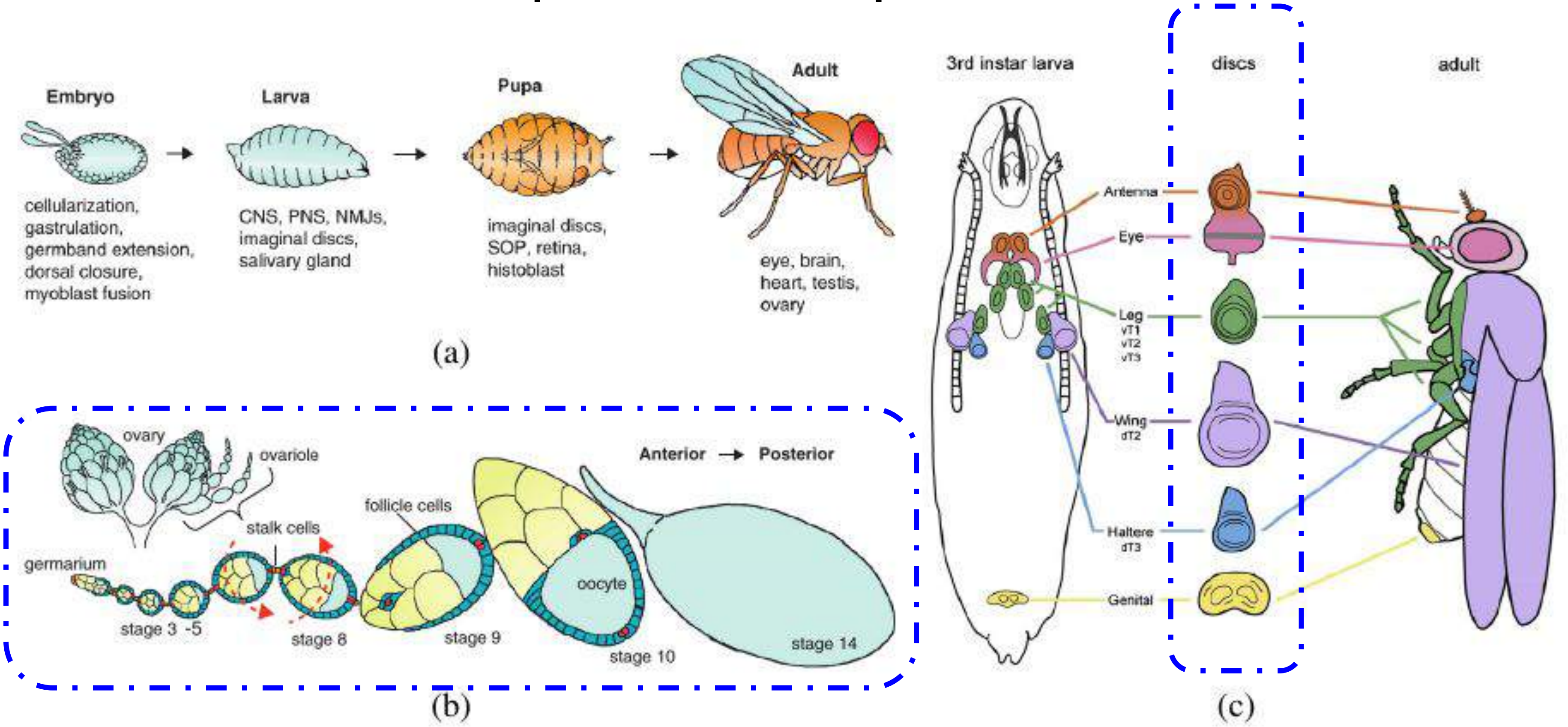
- High gene similarity with mumals (~ 93%)
- Short life cycles -> gene evaluation in generations

Automatic image processing:

- Analyses require thousands of images to be processed
- Very time consuming for experts



Schema of Drosophila development



Sample images

Drosophila ovaries

Imaginal disk

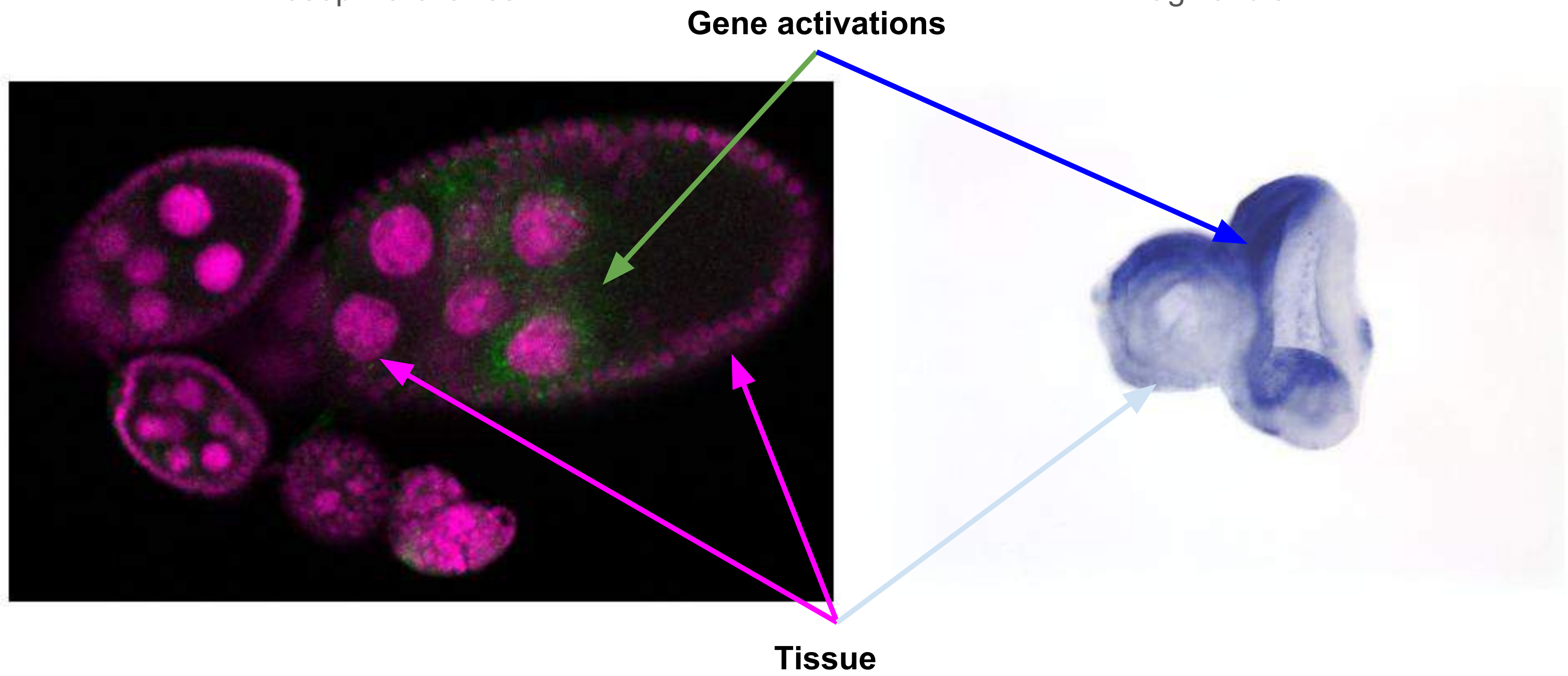
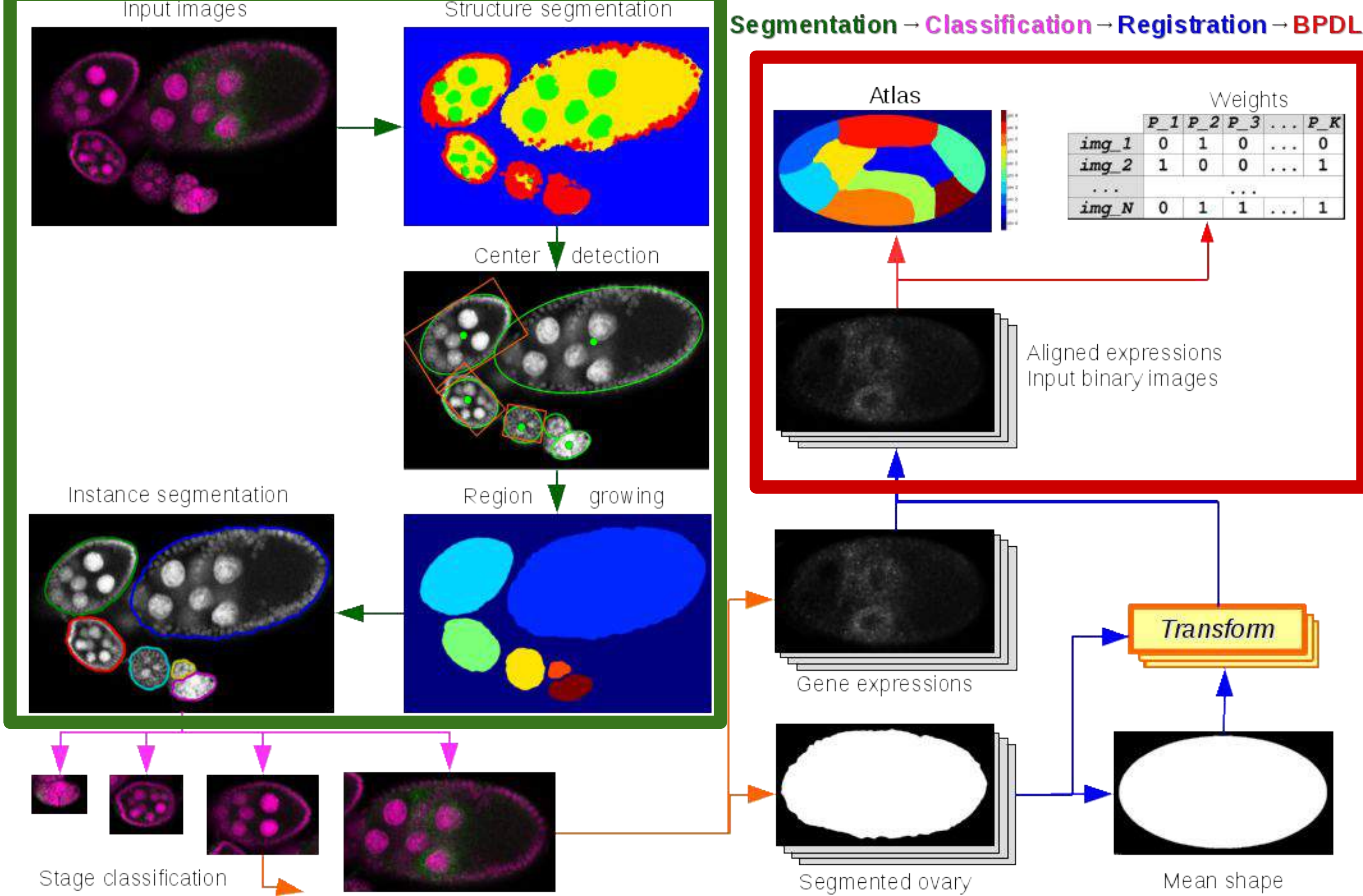


Image analysis pipeline

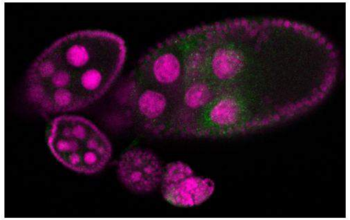


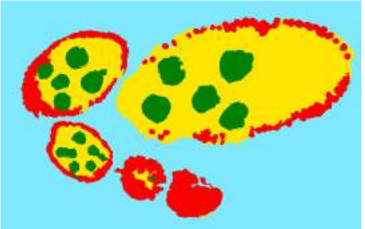
Drosophila Datasets

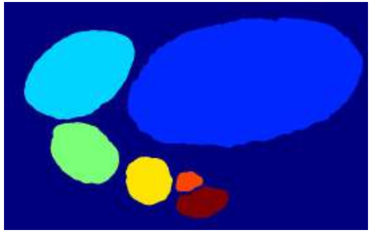
Just a few annotations in biomedical imaging is common issue…

- Noisy images with local deformations
- Low image variance (patterns)

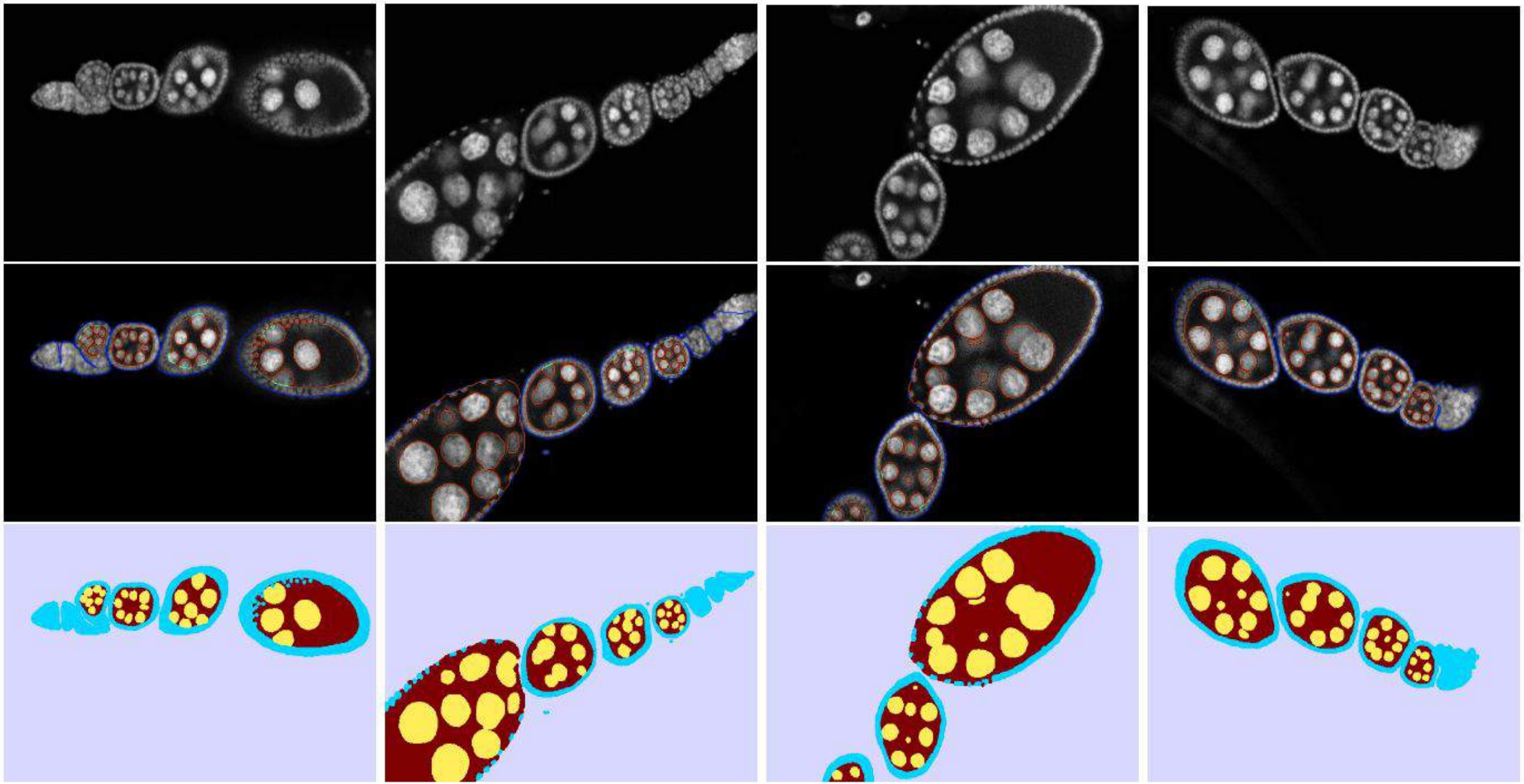
	Domain	Semantic annot.	Instance annot.
Ovaries	2.5D	72	250
Imaginal discs	2D	15	-



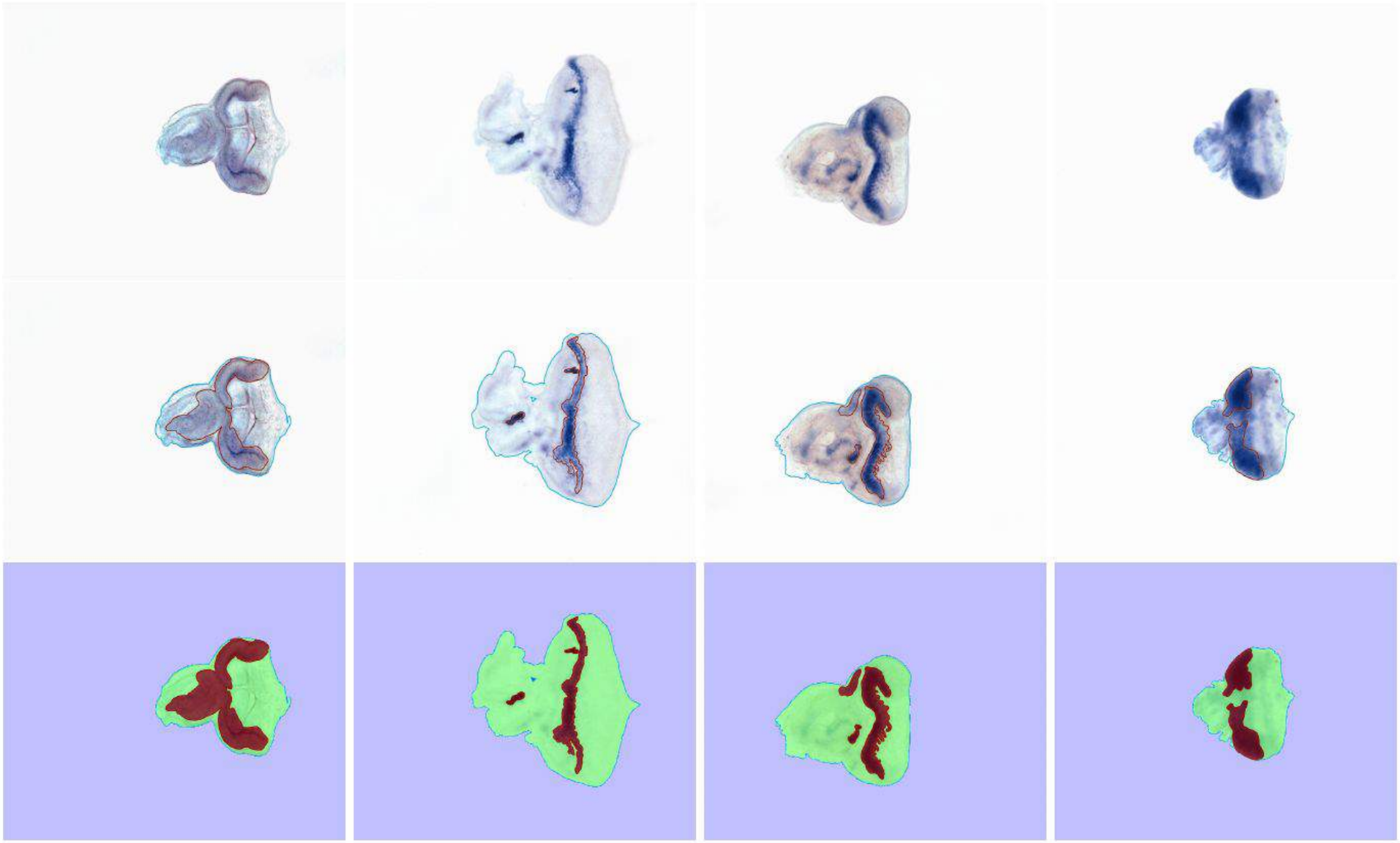




Drosophila ovary - semantic segmentation



Drosophila imaginal discs - semantic segmentation



Notations

Image related

Ω set of pixels (image plane)

I input image function $I : \Omega \rightarrow Rm$

L set of labels

Superpixels

~~η original regularization~~

~~ξ proposed regularization~~

~~v initial superpixel size~~

~~s superpixel~~

~~S set of superpixels $s \in S$~~

~~D weighted distance~~

~~$d_{\{c,s\}}$ color and spatial distance respectively~~

~~Ω_s pixels belonging to a superpixel s, $\Omega s \subset \Omega$~~

Segmentation & Region growing

y_Ω pixel-wise segmentation function $y\Omega : \Omega \rightarrow L$

Y_Ω ordered set of pixel-wise segmentation $Y\ \Omega = y\Omega\ (\Omega)$

y superpixel segmentation function $y : S \rightarrow L$ with abbrev. for $ys = y(s)$

Y ordered set of superpixel segmentation $Y = y(S)$

x feature vector

X set of features $xs \in X$ for all superpixels

$d_{\{M,E,T\}}$ Manhattan, Euclidean and Tchebychev distances

U, B unary (data) and binary (pairwise) term for GC respectively

g image descriptor

r vector of ray distances

c vector of object centers

m statistical shape model $m = [c, r, \Theta, w]$

M mixture of shape model

w vector with model weights

q cumulative probability of spatial prior

Dictionary Learning

g image appearance (binary association to a class)

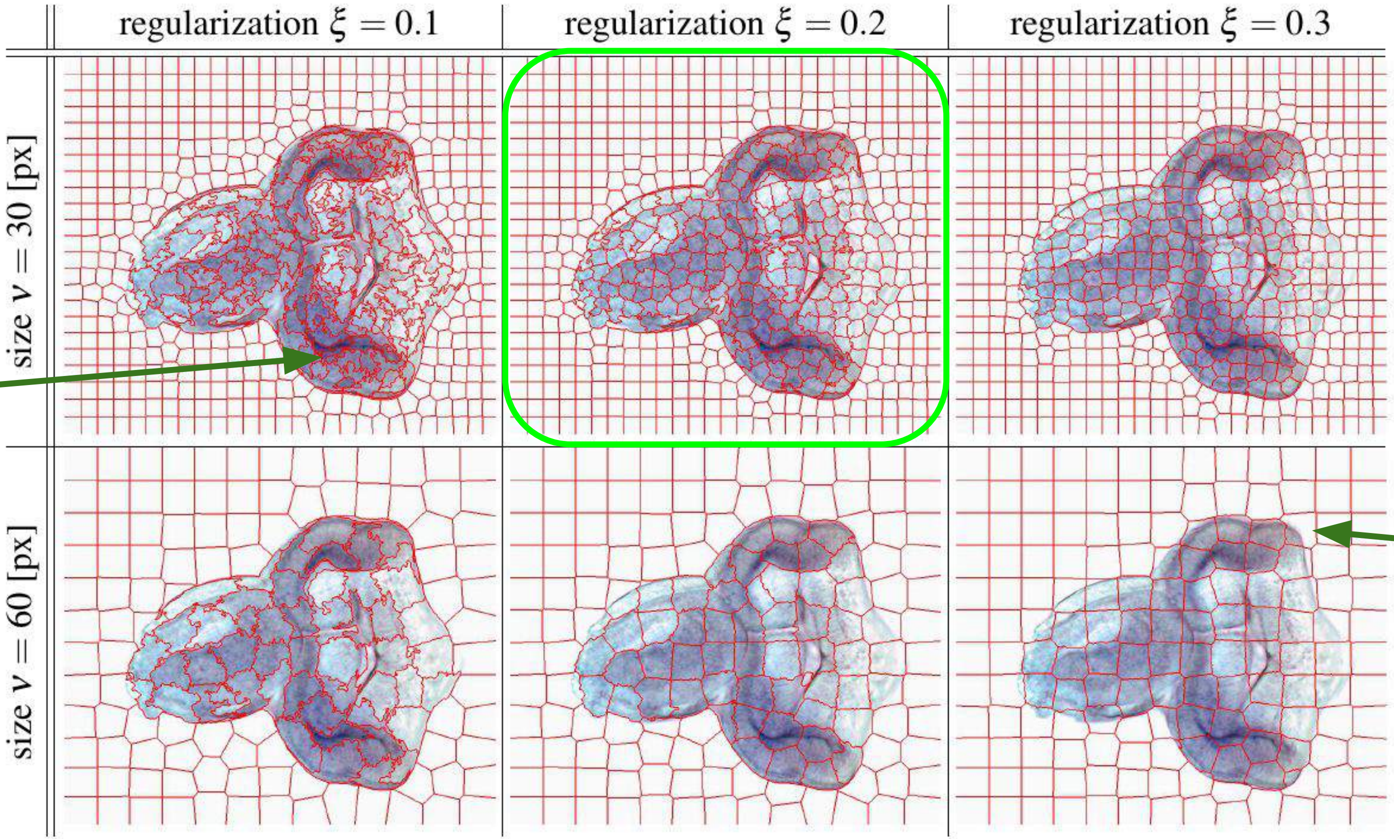
G set of images $g \in G$

y_Ω , Y_Ω atlas (binary patterns, segmentation)

w vector with binary weights

W matrix with binary weights $w \in W$

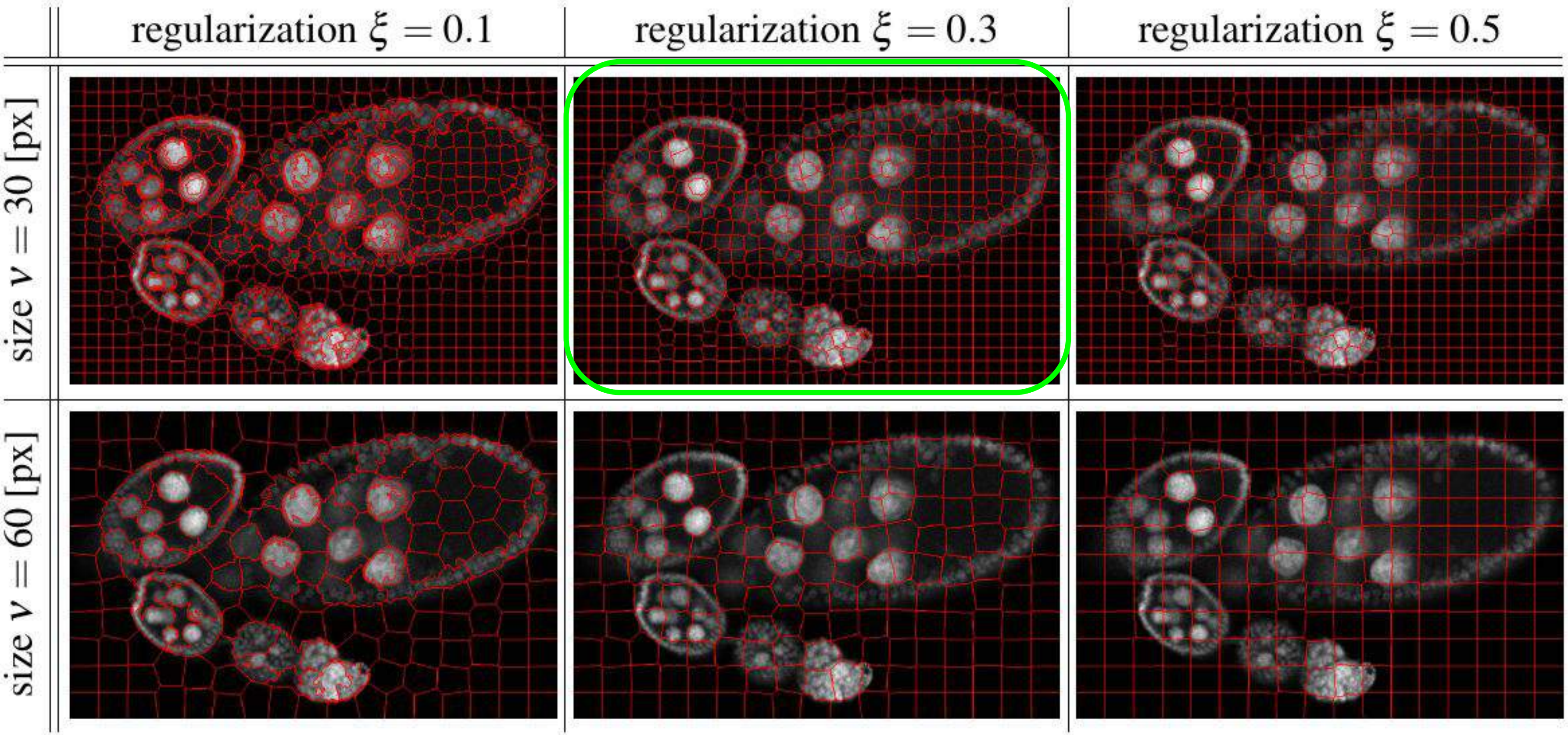
Superpixels - Imaginal discs



Too small & broken

Not reflecting object boundary

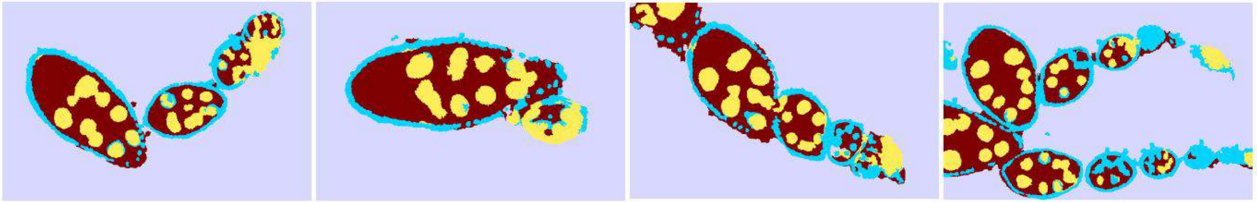
Superpixels - Drosophila ovary



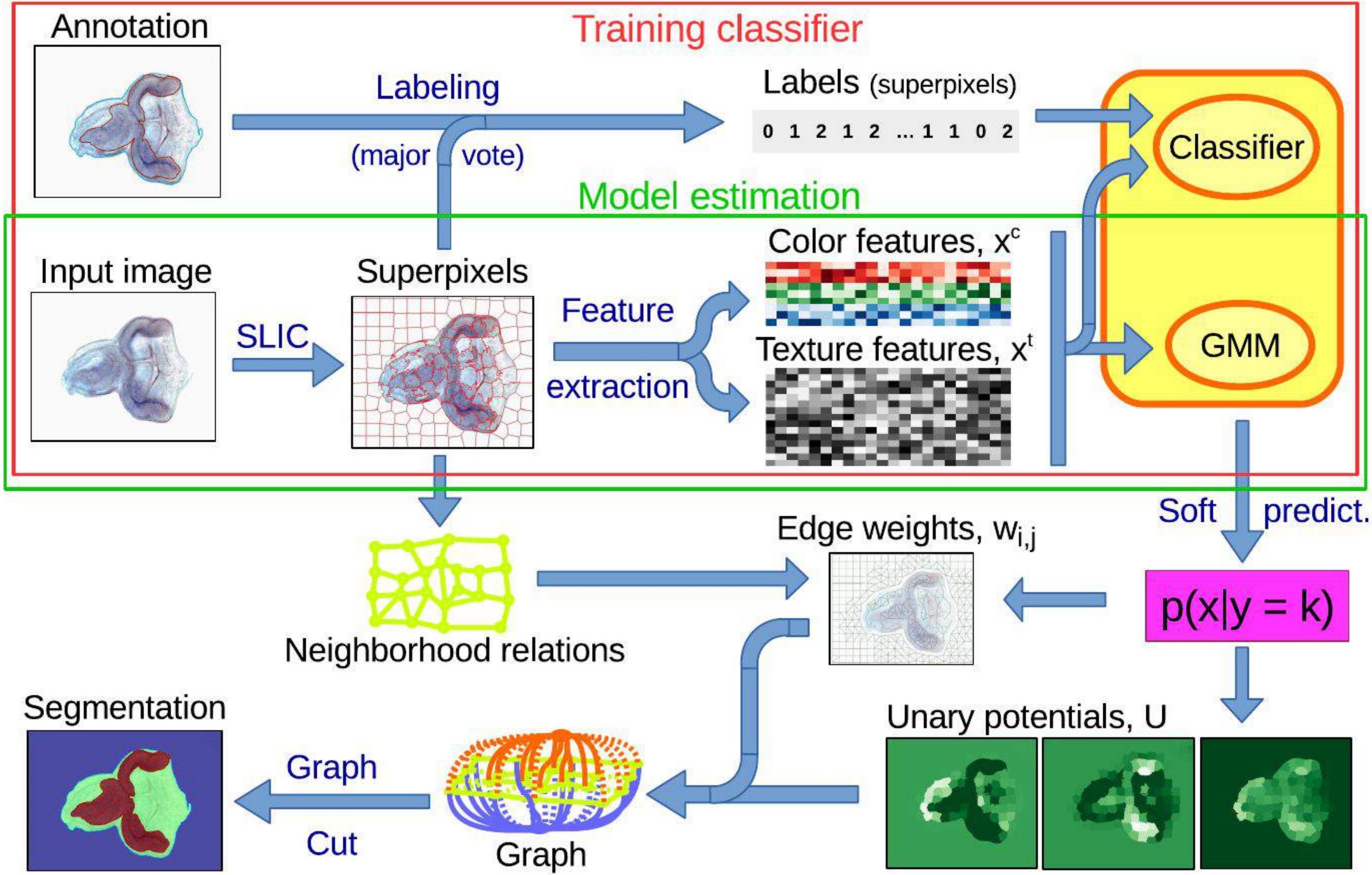
Overview

1. Problem statement
2. Used datasets
3. Instance segmentation
    a. Structure (tissue) segmentation on superpixels
    b. Center detection and ellipse fitting
    c. Region growing on superpixels
4. Binary Pattern Dictionary Learning
5. Conclusion & Future work

Supervised and unsupervised segmentation using superpixels, model estimation, and Graph Cut



Jiří Borovec, Jan Švihlík, Jan Kybic, David Habart, “**Supervised and unsupervised segmentation using superpixels, model estimation, and graph cut,**” Journal Electron. Imaging 26(6), 061610 (2017), [DOI: 10.1117/1.JEI.26.6.06161](DOI: 10.1117/1.JEI.26.6.06161)0.



Training classifier
Annotation
Labeling
(major vote)
Labels (superpixels)
0 1 2 1 2 ... 1 1 0 2
Classifier
Model estimation
Input image
SLIC
Superpixels
Feature extraction
Color features, x^c
Texture features, x^t
GMM
Soft predict.
Edge weights, $w_{i,j}$
$p(x|y = k)$
Neighborhood relations
Segmentation
Unary potentials, U
Graph Cut
Graph

Segmentation method overview

Image segmentation method consisting of the following steps:

1. Computation of superpixels - SLIC
2. Extraction of superpixel-based descriptors:
   a. Color - mean, median, energy, STD
   b. Texture - Leung-Malik filter bank
3. Calculating image-based class probabilities:
   a. Supervised - Random Forest, k-NN, Adaboost, ...
   b. Unsupervised - Gaussian Mixture Model
4. Spatial regularized superpixel classification using Graph Cut
   a. Edge weights - color, features, model

Problem formulation

Formulation (standard) as

$$Y^* = \arg\max_Y P(Y|X) = \arg\max_Y \frac{p(X|Y) \cdot P(Y)}{p(X)}$$

$$P(Y) = \prod_{s \in S} h(y_s) \cdot \prod_{(i,j) \in \mathcal{N} \subseteq S^2} R(y_i, y_j)$$

$$Y^* = \arg\max_Y \prod_{i \in S} \left(p(\boldsymbol{x}_i|y_i) \cdot h(y_i)\right) \cdot \prod_{(i,j) \in \mathcal{N}} R(y_i, y_j)$$

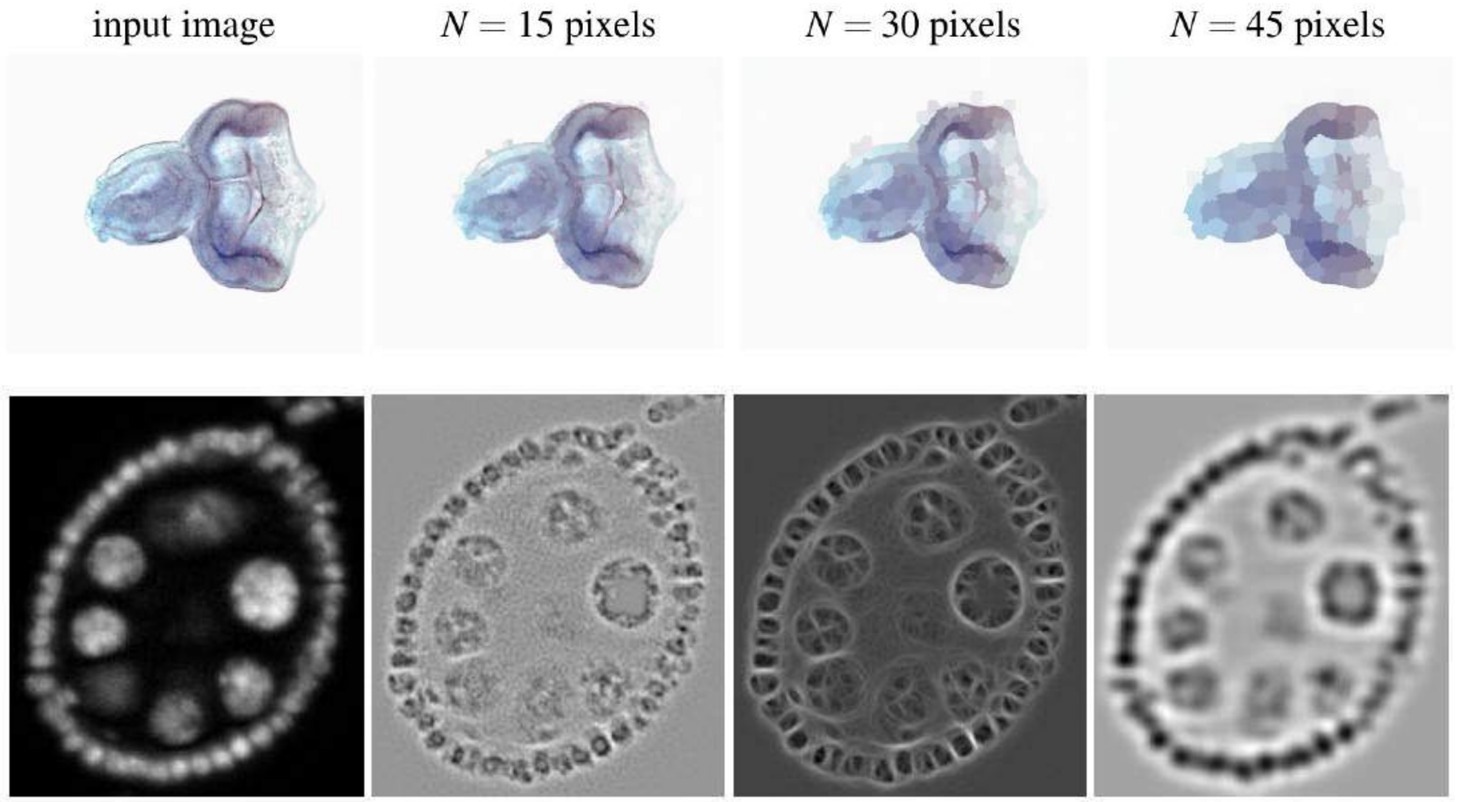
Energy minimisation

$$Y^* = \arg\min_Y \sum_s \underbrace{-\log\left(p(\boldsymbol{x}_s|y_s) \cdot h(y_s)\right)}_{U_s(y_s)} + \sum_{(i,j) \in \mathcal{N}} \underbrace{-\log R(y_i, y_j)}_{\beta w_{i,j} B(y_i, y_j)}$$

Superpixel features

Superpixel color means

Leven-Malik filter bank

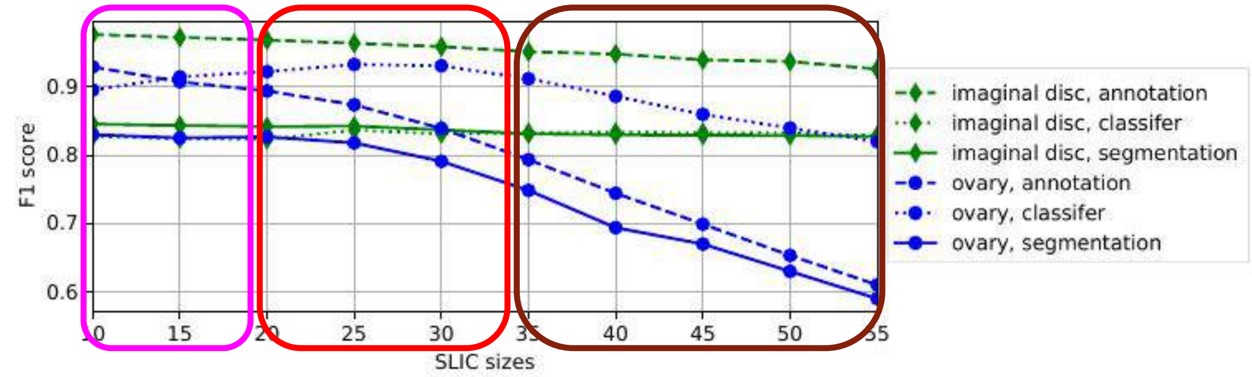


Influence of superpixel parameters

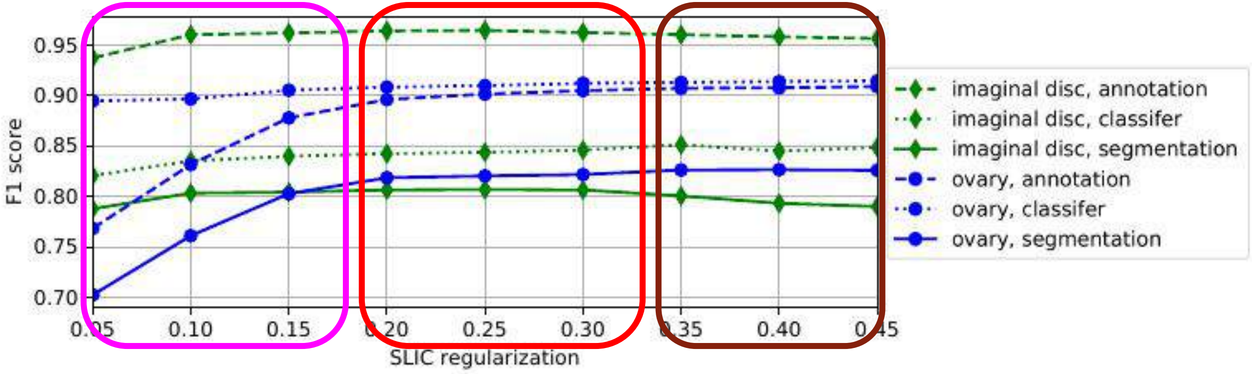
Low feature representativeness

Optimum (~compromise)

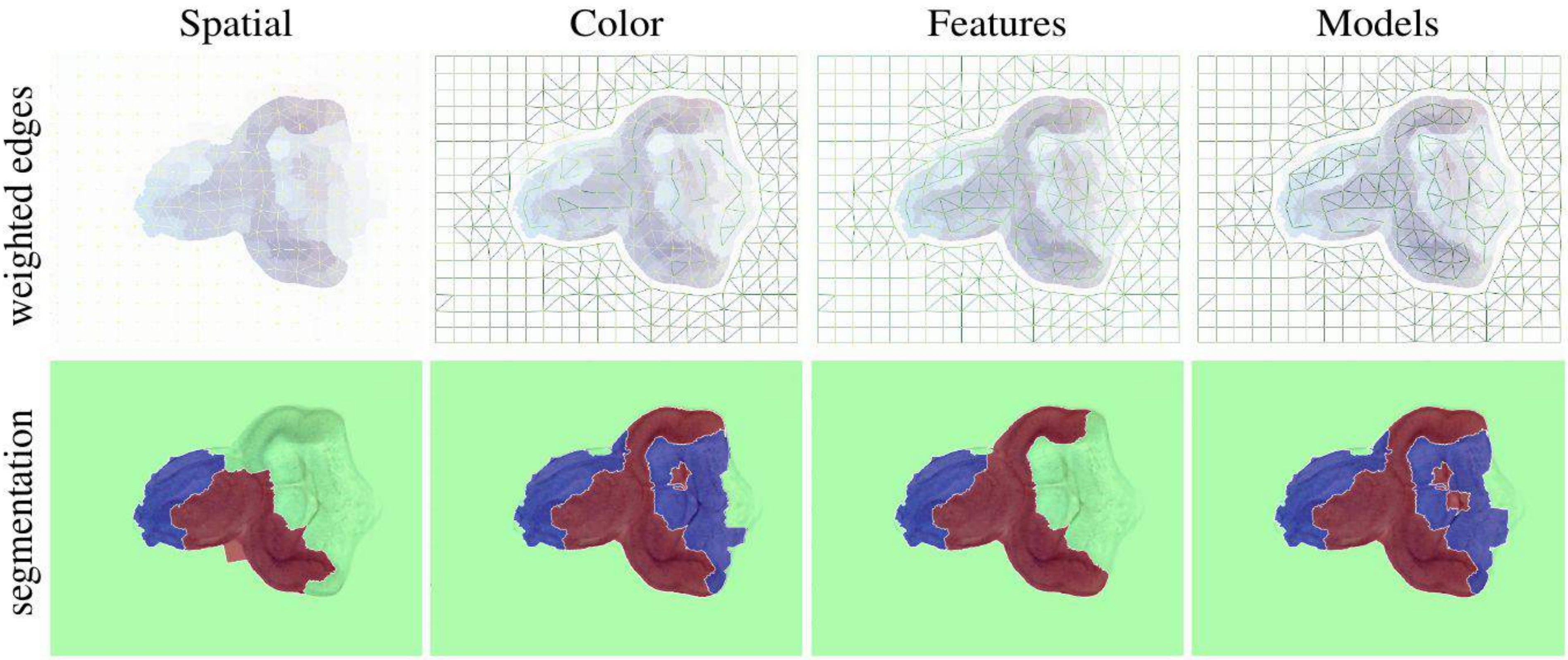
Low separability



(a)



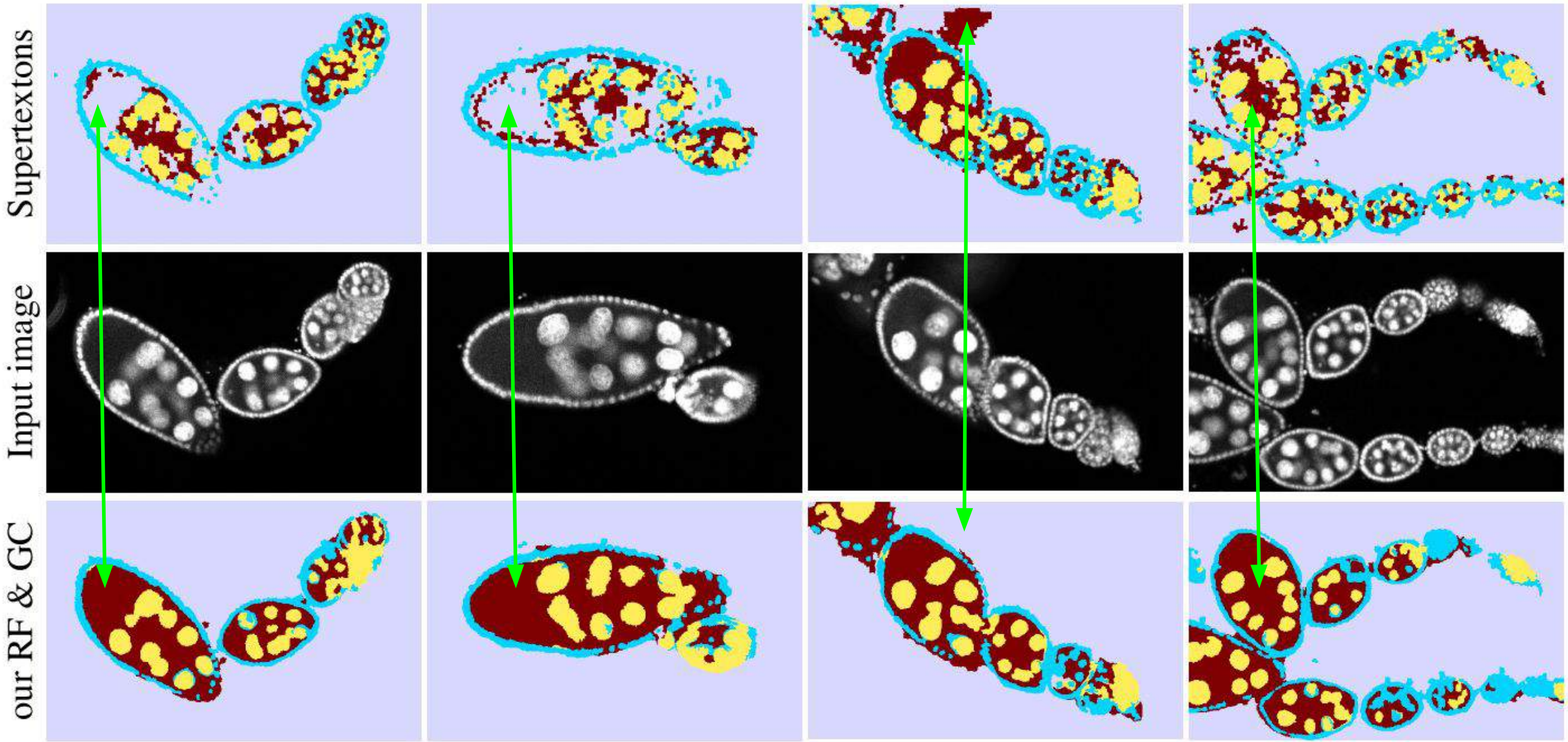
Graph Cut - Edge weights



Segmentation results with SOA (F1-score)

		Method	imaginal disc	ovary
Pixel-wise	Supervised	Weka	0.6923	0.5800
Pixel-wise	Supervised	Weka & GC(0, 100)	0.6887	0.5810
Pixel-wise	Supervised	Weka & GC(1, 50)	0.6887	0.5965
Pixel-wise	Supervised	Weka & GC(10, 50)	0.6887	0.1395
Pixel-wise	Supervised	Weka & GC(50, 100)	0.6850	0.6007
Superpixels	Supervised	ideal segm. Y_A	0.9696	0.9067
Superpixels	Supervised	Supertextons	-	0.7488
Superpixels	Supervised	our RF	0.8181	0.8201
Superpixels	Supervised	our RF & GC	**0.8229**	**0.8600**
Superpixels	Unsuper.	our GMM	0.7542	0.5967
Superpixels	Unsuper.	our GMM & GC	*0.7644*	0.6039
Superpixels	Unsuper.	our GMM [gr]	0.7301	0.6009
Superpixels	Unsuper.	our GMM [gr] & GC	0.7564	*0.6083*

Advantage of using Graph Cut

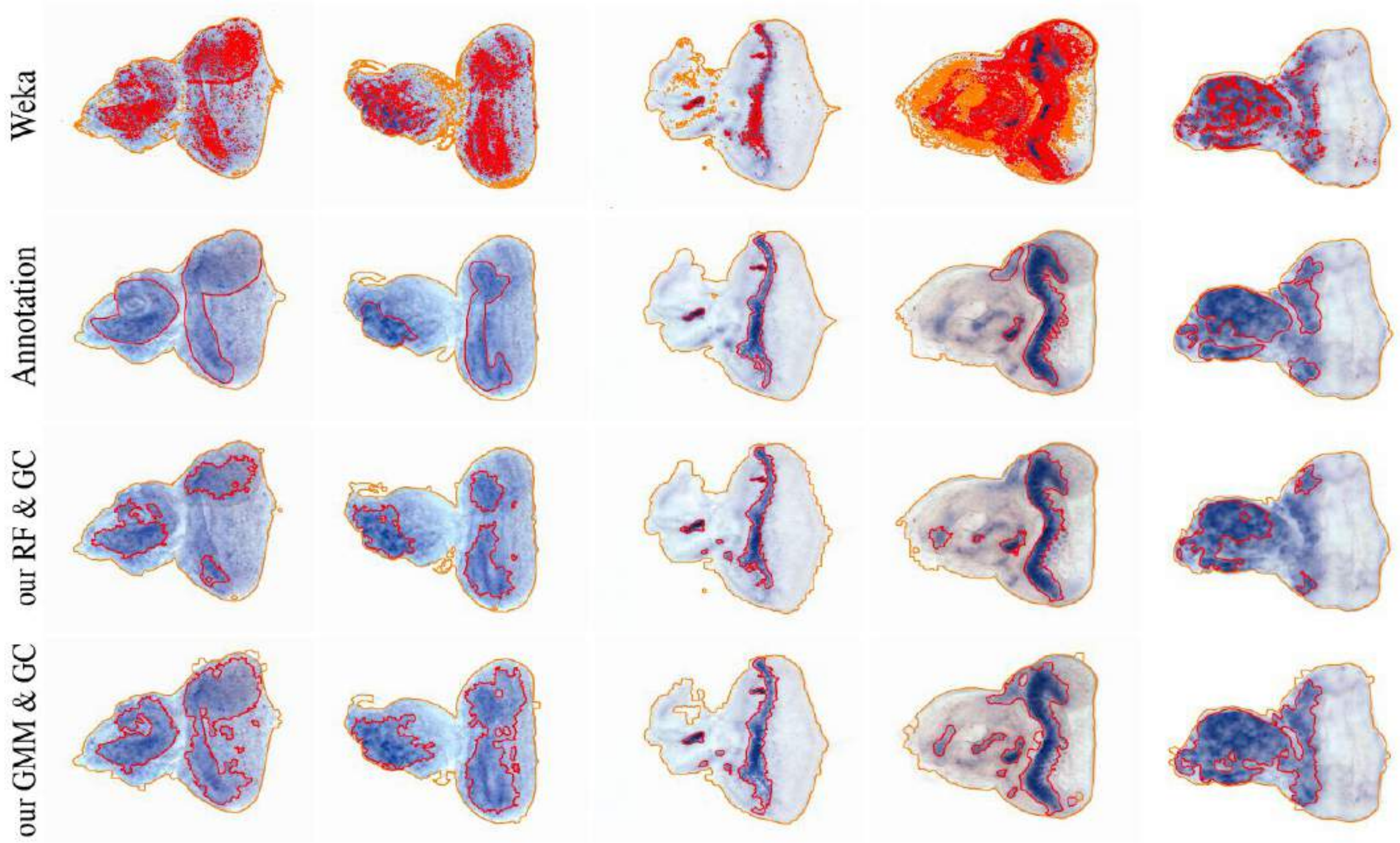


Supervised vs Unsupervised

Supervised

vs.

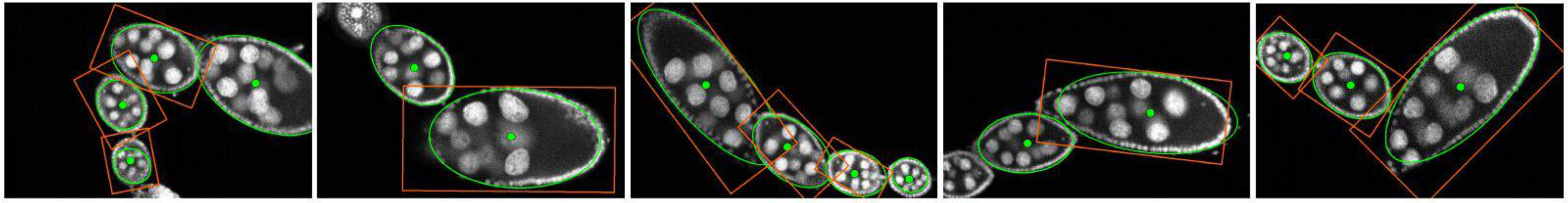
Unsupervised



Overview

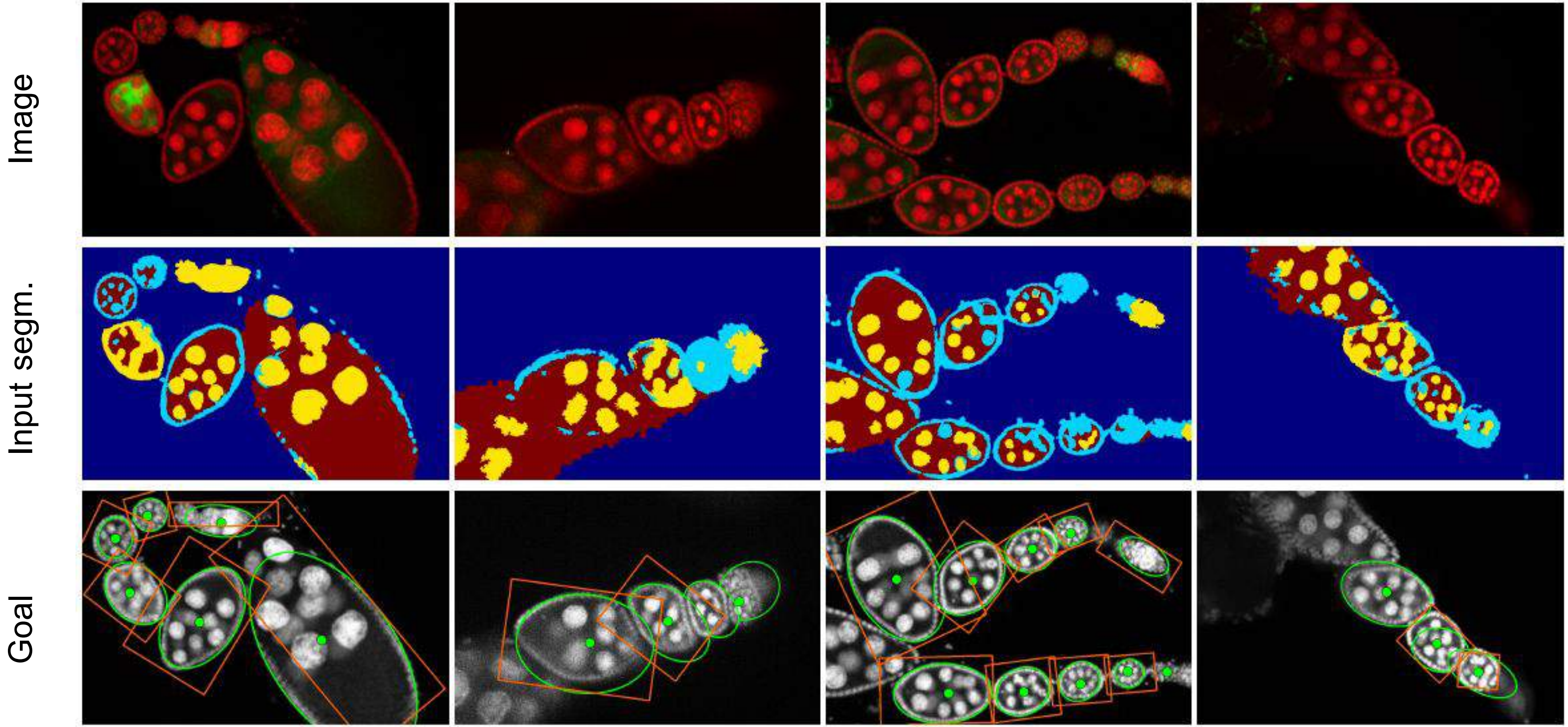
1. Problem statement
2. Used datasets
3. Instance segmentation
    a. Structure (tissue) segmentation on superpixels
    b. Center detection and ellipse fitting
    c. Region growing on superpixels
4. Binary Pattern Dictionary Learning
5. Conclusion & Future work

Detection and localization of Drosophila egg chambers in microscopy images



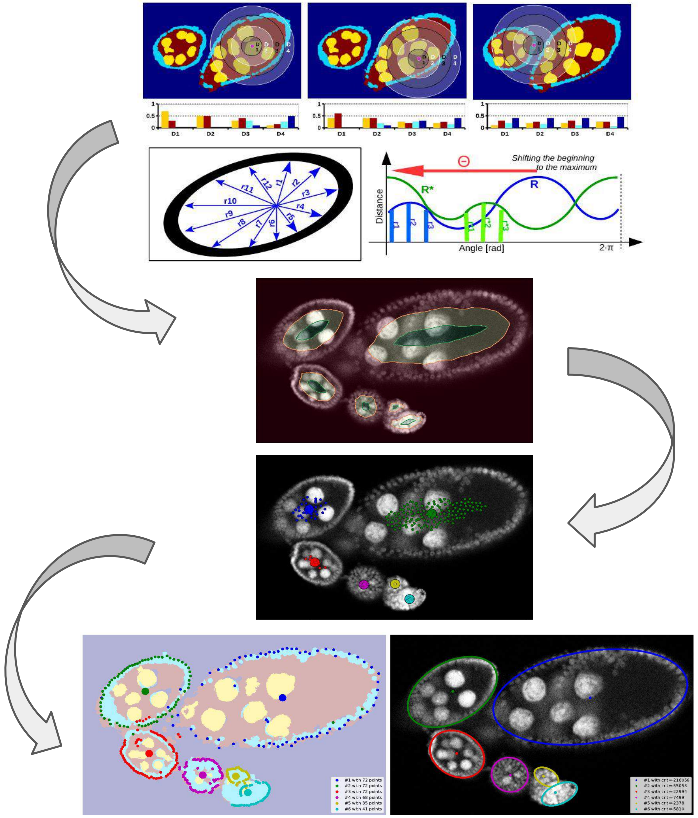
Borovec J., Kybic J., Nava R. (2017) **Detection and Localization of Drosophila Egg Chambers in Microscopy Images**. In: Machine Learning in Medical Imaging. LNCS, vol 10541. Springer, [DOI: 10.1007/978-3-319-67389-9_3](https://doi.org/10.1007/978-3-319-67389-9_3).

Center detections - illustration



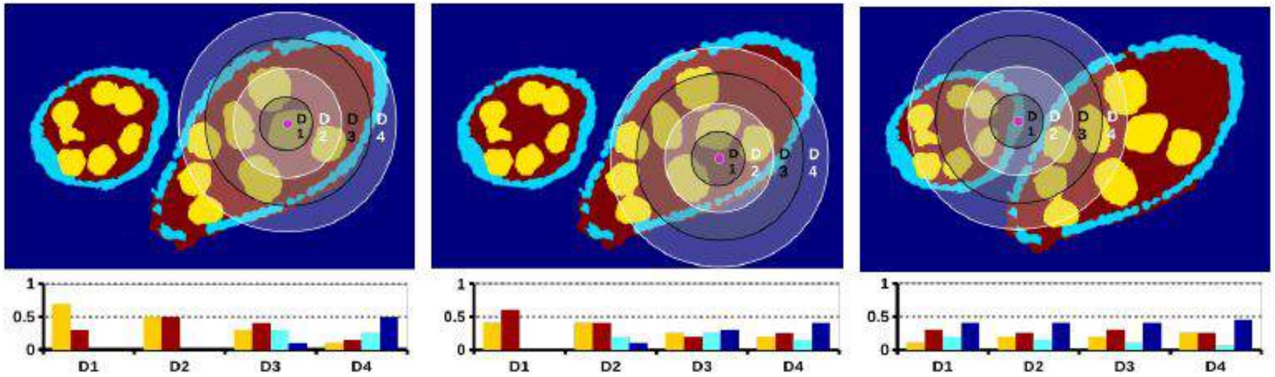
Schema

1. Extract pixel features
   a. Label histogram
   b. Ray features
2. Train classifier
3. Group center candidates
4. Ellipse fitting

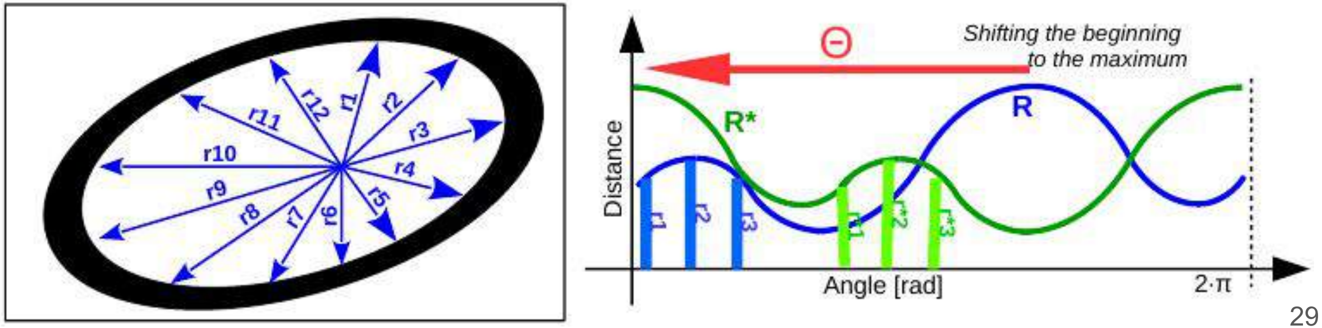


Features for center detection

- Label histogram



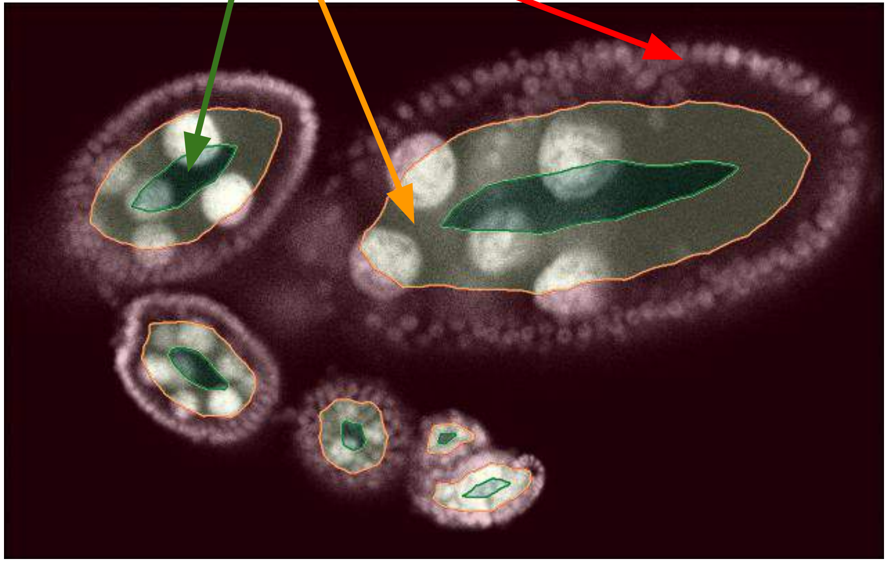
- Ray features



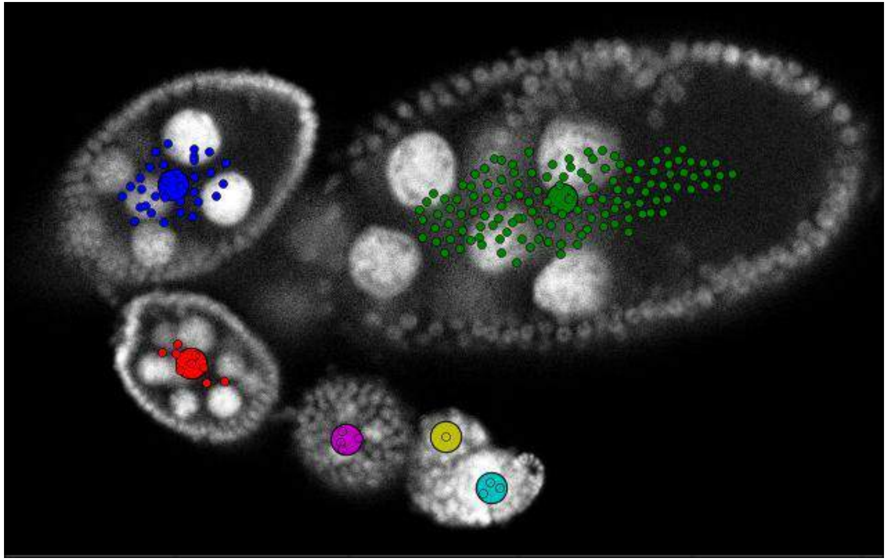
Classification & Grouping

Train classifier on 3 zones

- Negative
- ~~No not case~~
- Positive

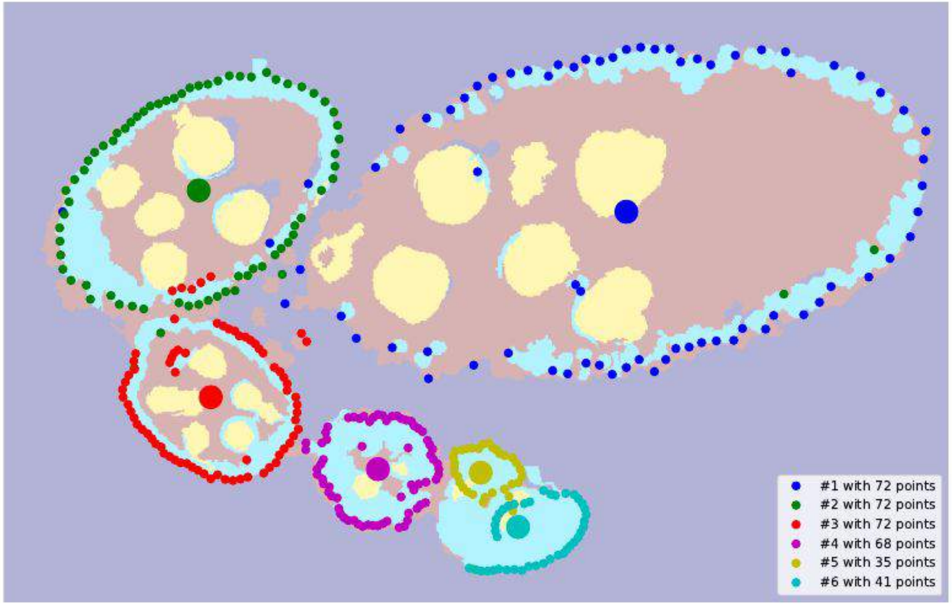


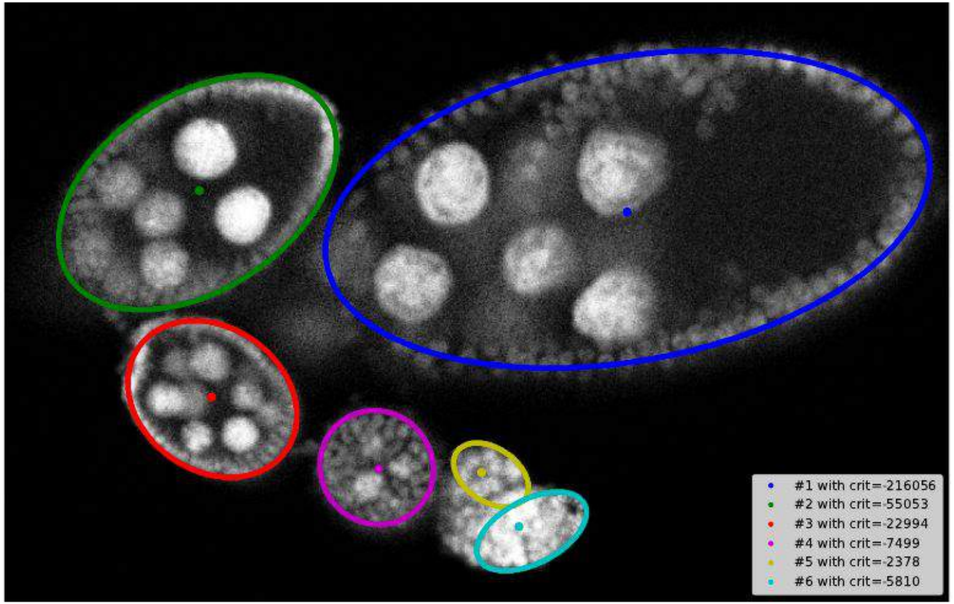
Clustering of center candidates with DBSCAN



Ellipse fitting

Fitting ellipses to boundary points to maximise foreground labels inside ellipse





Ellipse fitting

Formulate as Maximize likelihood

$$\arg\max_{F,B} \prod_{i\in\Omega_F} P_F(Y_i) \cdot \prod_{i\in\Omega\setminus\Omega_F} P_B(\mathbf{y}_i)$$

$$h_\bullet = -\log P_\bullet$$

$$\sum_{i\in\Omega} h_B(Y_i) = \sum_{i\in\Omega_F} h_B(Y_i) + \sum_{i\in\Omega\setminus\Omega_F} h_B(Y_i)$$

$$\arg\min_{F,B} \sum_{i\in\Omega_F} h_F(Y_i) + \sum_{i\in\Omega} h_B(Y_i) - \sum_{i\in\Omega_F} h_B(Y_i)$$

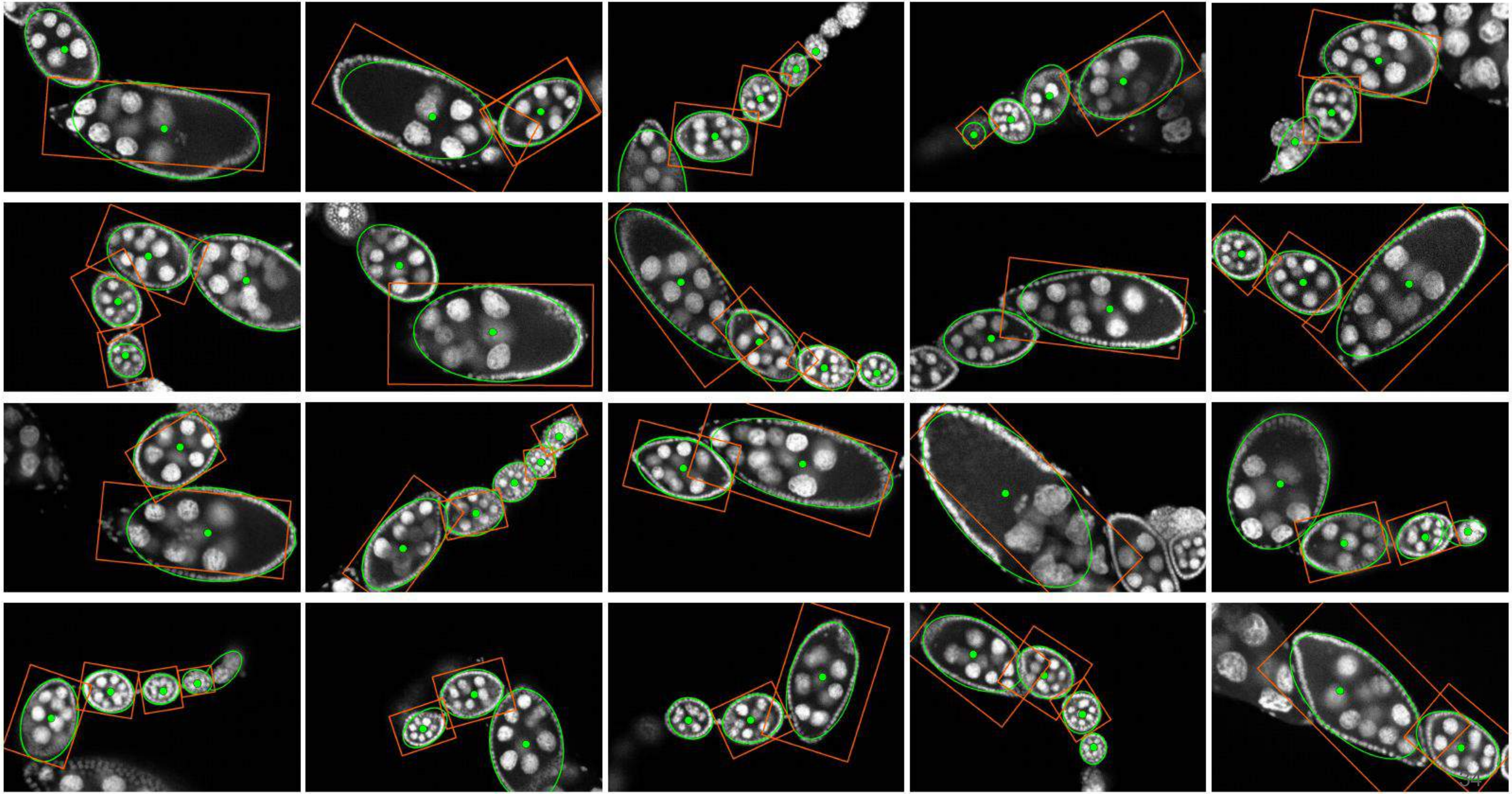
$\sum_{i\in\Omega} h_B(Y_i)$ is constant

$$\arg\min_{F,B} \sum_{i\in\Omega_F} h_F(Y_i) - h_B(Y_i)$$

Ellipse pruning

Ellipse approximation eliminate multiple center detection in single egg

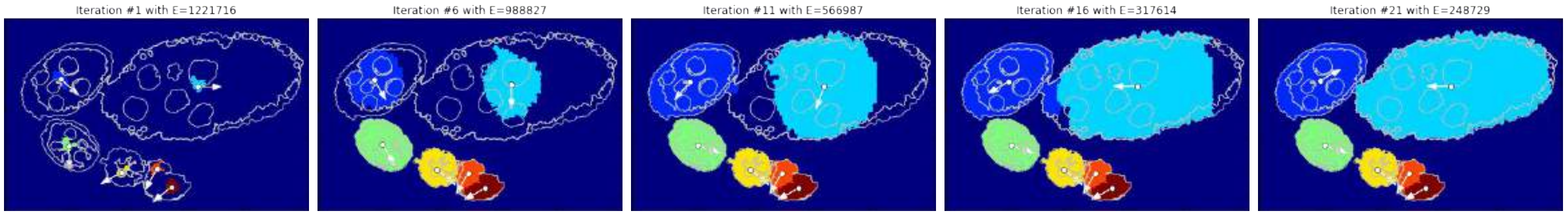
Egg chambers	Stage				
	1	2	3	4	5
number	921	1403	865	834	836
false negatives	306 (33%)	158 (11%)	6 (0.7%)	1 (0.1%)	0 (0.0%)
multiple detections (MD)	37 (4.0%)	31 (2.2%)	109 (12%)	80 (9.6%)	90 (11%)
MD after ellipse fitting	18 (2.0%)	13 (0.9%)	27 (3.1%)	20 (2.4%)	30 (3.6%)
false positives	43 (0.9%)				



Overview

1. Problem statement
2. Used datasets
3. Instance segmentation
    a. Structure (tissue) segmentation on superpixels
    b. Center detection and ellipse fitting
    c. Region growing on superpixels
4. Binary Pattern Dictionary Learning
5. Conclusion & Future work

Region growing using superpixels with learned shape prior



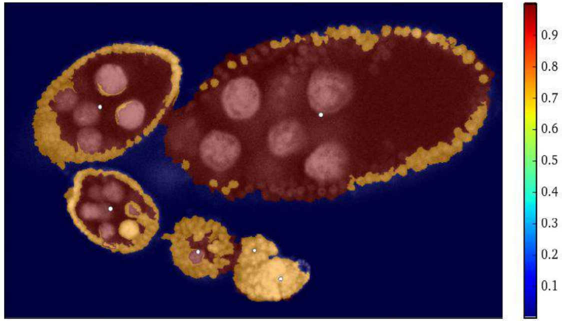
Jiří Borovec, Jan Kybic, Akihiro Sugimoto, "**Region growing using superpixels with learned shape prior,**" Journal Electron. Imaging 26(6), 061610 (2017), [DOI: 10.1117/1.JEI.26.6.06161](https://doi.org/10.1117/1.JEI.26.6.061611)1.

Region growing - variational framework

Formulated as:

$$P(\mathbf{y}_s \mid y, M) = \frac{1}{Z(M,y)} \boxed{P_g(g \mid y)}\, \boxed{P_m(g \mid M)}\, \boxed{P_R(g)}$$

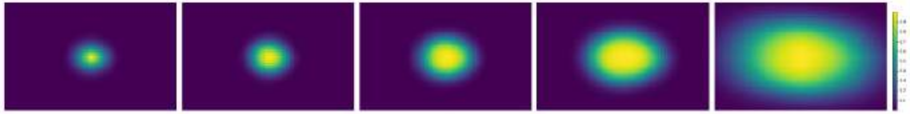
Appearance model



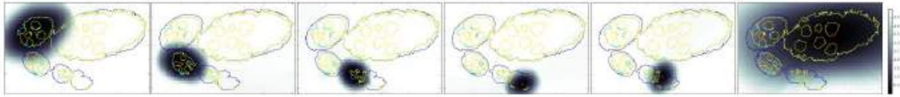
Pairwise term

$$P_R(g) = \prod_{(u,v)\in\mathcal{N}_S} H\big(\mathbf{y}(u), \mathbf{y}(v)\big)$$

Mixture model



(b)



(c)

Region growing - variational framework

Where:

$$P_g(g \mid y) = \prod_{i \in \Omega} P_g\Big(g\big(s(i)\big) \mid y\big(s(i)\big)\Big) = \prod_{s \in S} P_g\big(\mathbf{y}_s \mid y(s)\big)^{|\Omega_s|}$$

$$P_m(g \mid M) = \prod_{i \in \Omega} P_m\Big(g\big(s(i)\big) \mid M\Big) = \prod_{s \in S} P_m\Big(\mathbf{y}_s \mid M\Big)^{|\Omega_s|}$$

$$P_R(g) = \prod_{(u,v) \in \mathcal{N}_S} H\big(\mathbf{y}(u), \mathbf{y}(v)\big)$$

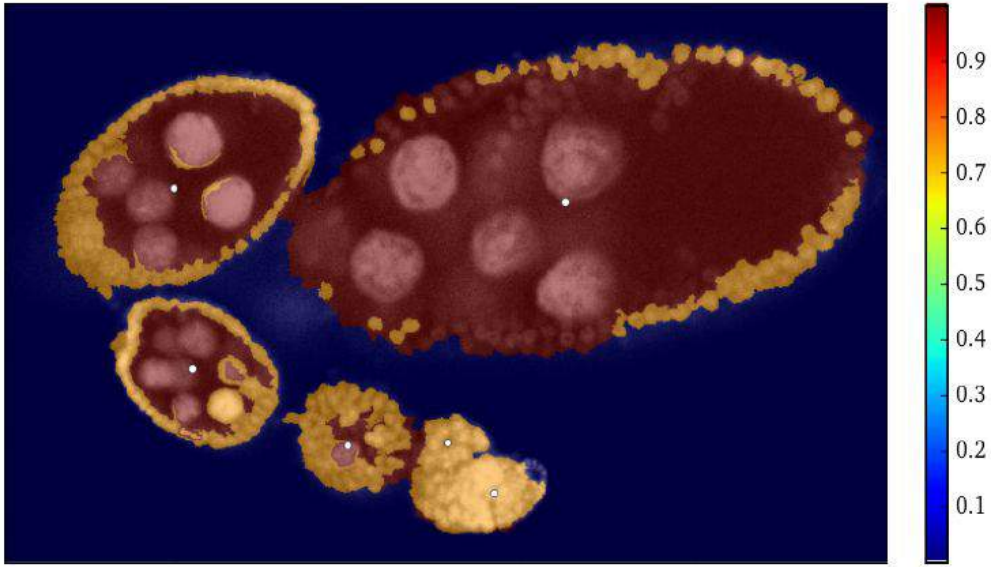
Resolves in energy minimisation:

$$E'(g) = \sum_{s \in S} \underbrace{|\Omega_s| \Big[D_s(\mathbf{y}_s) + \beta V_s(\mathbf{y}_s)\Big]}_{U_s(y_s)} + \gamma \sum_{(u,v) \in \mathcal{N}_S} B\big(\mathbf{y}(u), \mathbf{y}(v)\big)$$

Appearance model

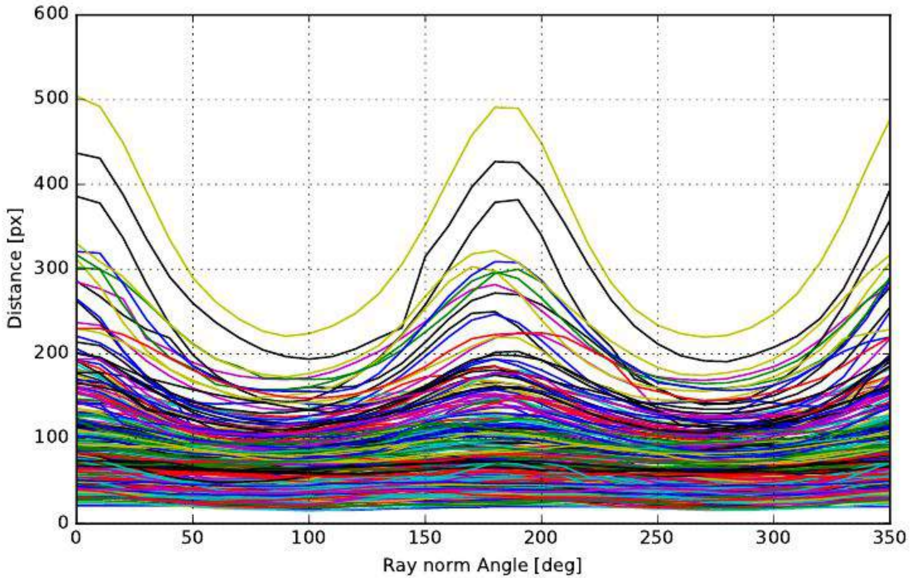
Associating a probability for each pixel (superpixel) whether it belongs to an object or not

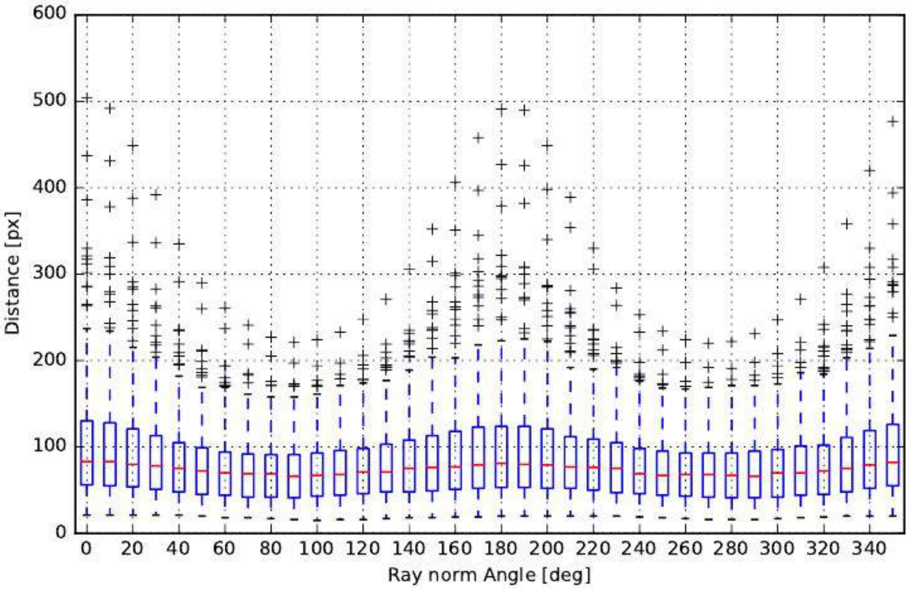
$$P_g\big(\mathbf{y}(s)|g_s\big) = \begin{cases} P_g(g_s) & \text{for } \mathbf{y}(s) \neq 0 \\ 1 - P_g(g_s) & \text{for } \mathbf{y}(s) = 0 \end{cases}$$



Measured shapes

Measured egg shapes as ray features from training examples (~250 eggs)



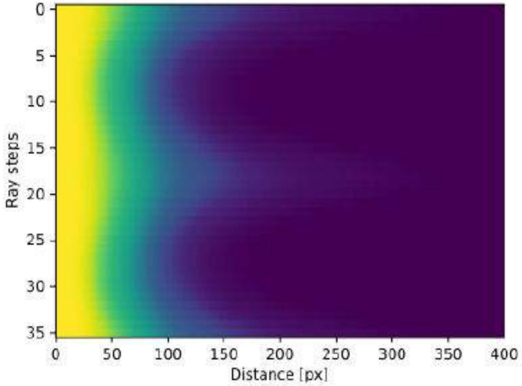


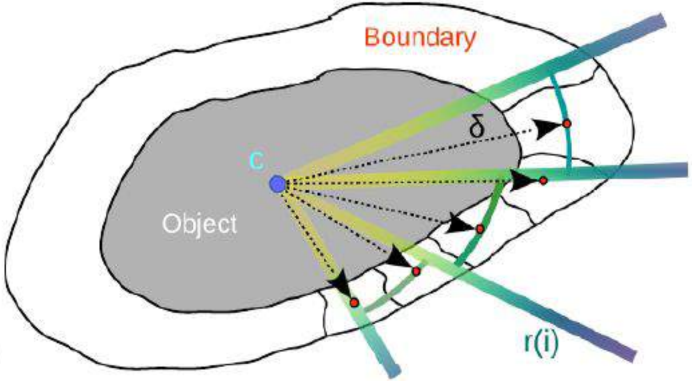
Shape model & prior

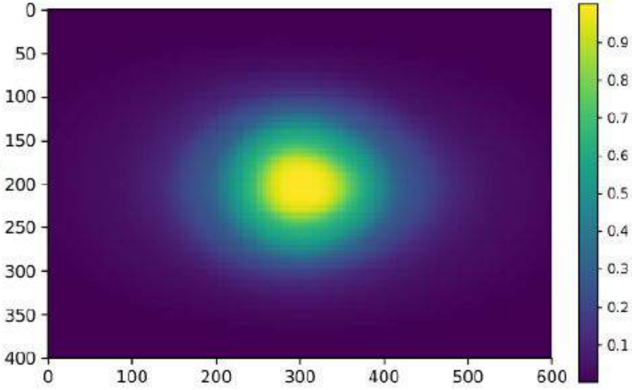
Gaussian model from all measured Ray features

Prior is represented as integral of probabilities

$$q(s, m_k) = \int_{\delta}^{\infty} \rho(r)\,\mathrm{d}r = 1 - \int_{0}^{\delta} \rho(r)\,\mathrm{d}r$$

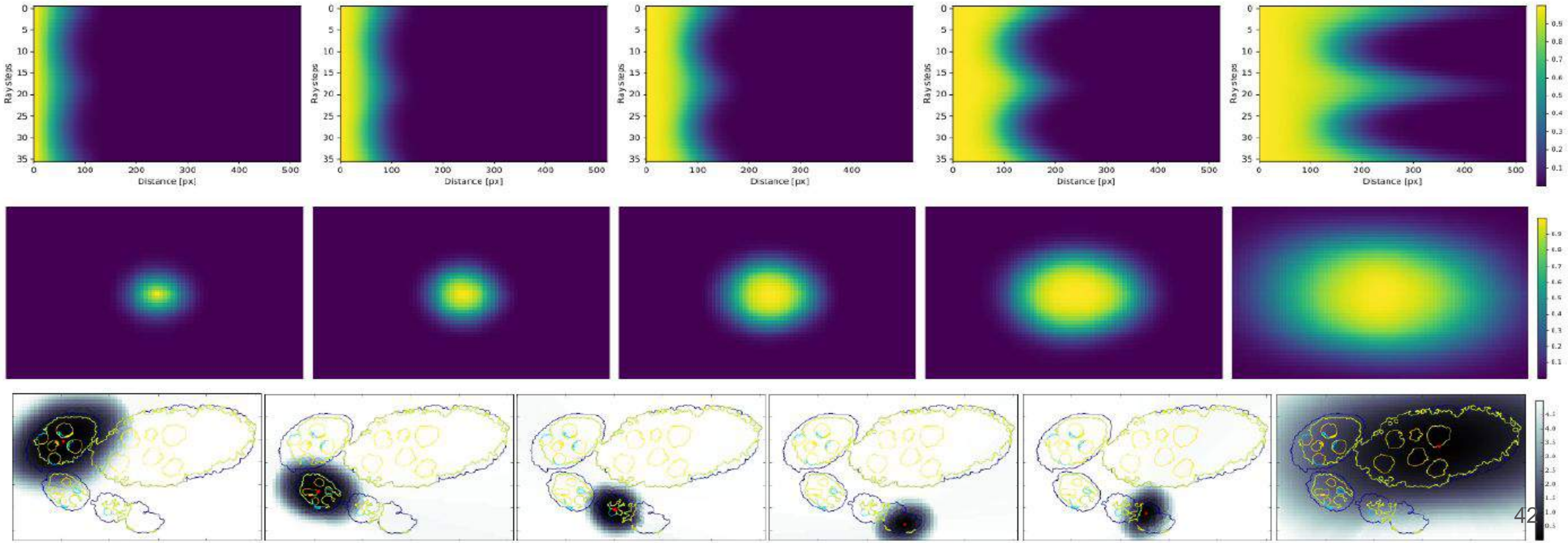






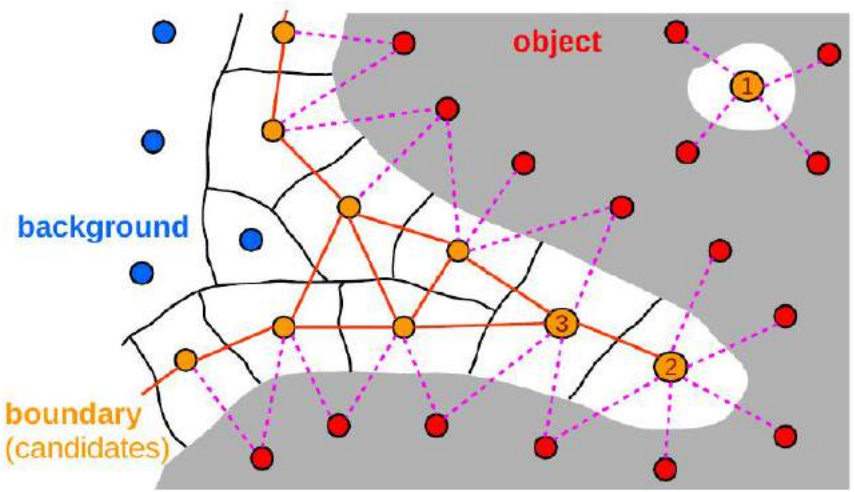
Mixture of Gaussian models

$$P_m(\mathbf{y}_s = k \mid M) = \begin{cases} q(s, m_k) & \text{for } k > 0 \\ \prod_l (1 - q(s, m_l)) & \text{for } k = 0 \end{cases}$$



RG - optimisation

- Iterative approach on object boundaries
- Alternating: region growing & update shape prior
- Strategies:
    - Greedy
    - Graph Cut
        - Binary
        - Multi-class
    - Object swapping

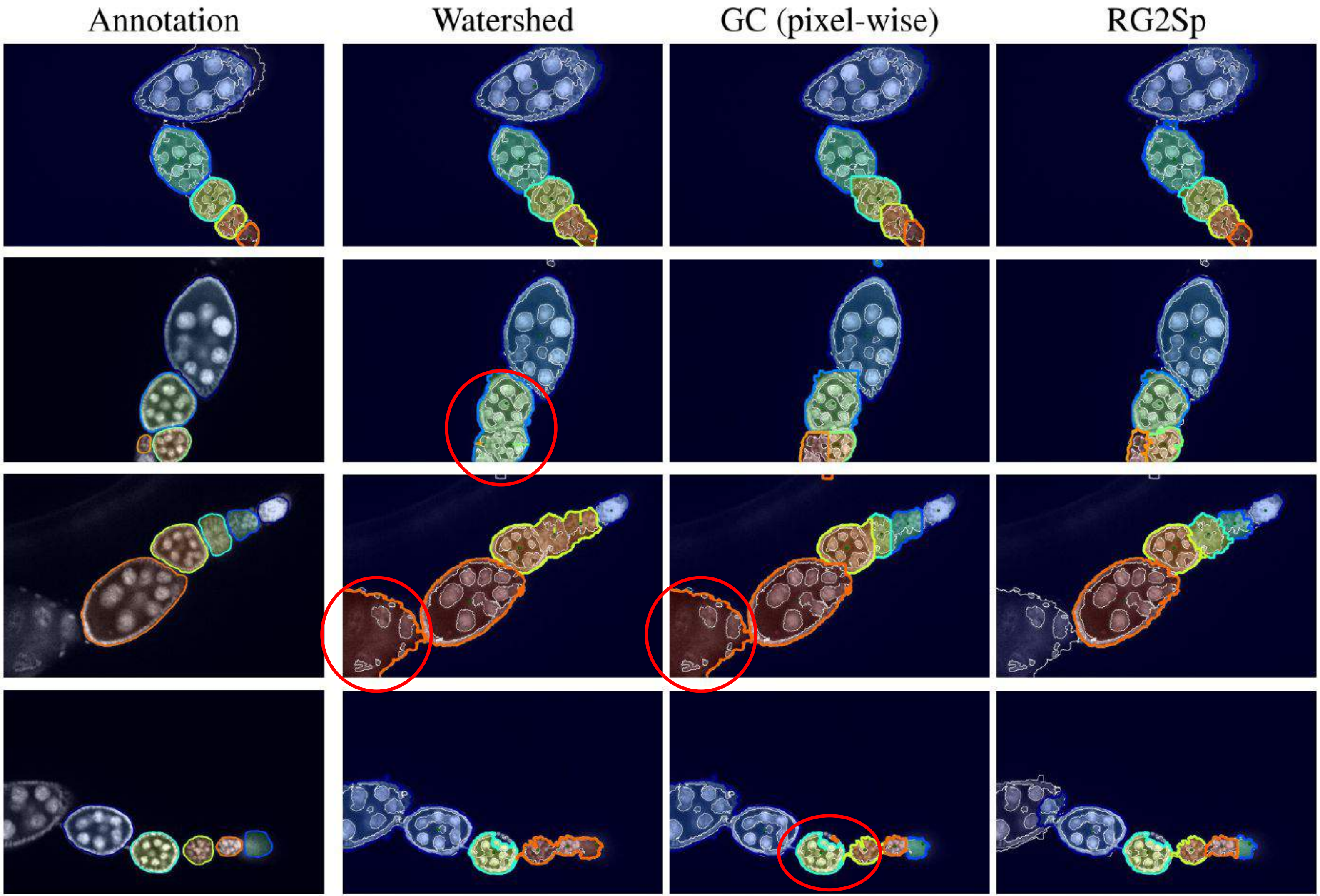


Algorithm 1: Region growing.

Input: S: superpixels, g: superpixel descriptors, c_k: initial object centers, M: mixture of statistical shape models

Output: object segmentation $\mathbf{y}$

compute data cost D;
from object centers c_k set initial segmentation $\mathbf{y}$ and model shape parameters m_k;
compute shape cost V;
while *not converged* **do**
    update object pose parameters c_k and Θ_k ;
    if *significant change of center c_k position, orientation Θ_k and object area* **then**
        update remaining object shape parameters m_k;
        update shape costs V for all s close to k;
    end
    find superpixels ∂S_k on the external object boundary of k;
    optimize (7.10) wrt $\mathbf{y}$ by changing $s \in \partial S_k$ using the greedy or Graph Cut algorithms;
end


Annotation
Watershed
GC (pixel-wise)
RG2Sp

Result compared to SOA

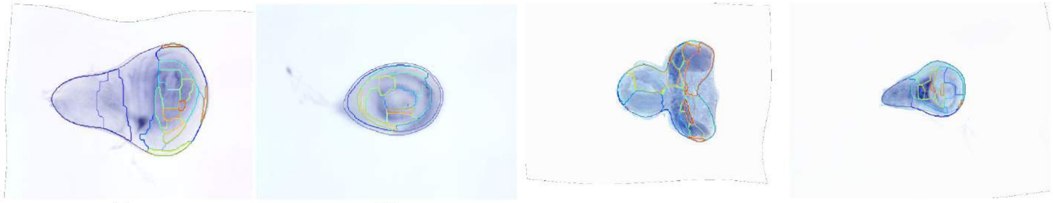
High Jaccard index with reasonable processing time

	Jaccard	accuracy	F_1 score	precision	recall	time [s]
Watershed	0.5705	0.9246	0.9246	0.9246	0.9246	5
Watershed (w. morph.)	0.5705	0.9270	0.9198	0.9136	0.9327	7
Morph. snakes (image)	0.4251	0.8769	0.8070	0.9053	0.7987	784
Morph. snakes (P_y)	0.6494	0.8812	0.8812	0.8812	0.8812	968
Graph Cut (pixel-level)	0.7143	0.9204	0.9204	0.9204	0.9204	15
Graph Cut (superpixels)	0.3164	0.8643	0.8643	0.8643	0.8643	3
RG2Sp (greedy)	0.7527	0.9577	0.9577	0.9577	0.9577	72
RG2Sp (Graph Cut)	0.7544	0.9568	0.9568	0.9568	0.9568	9

Overview

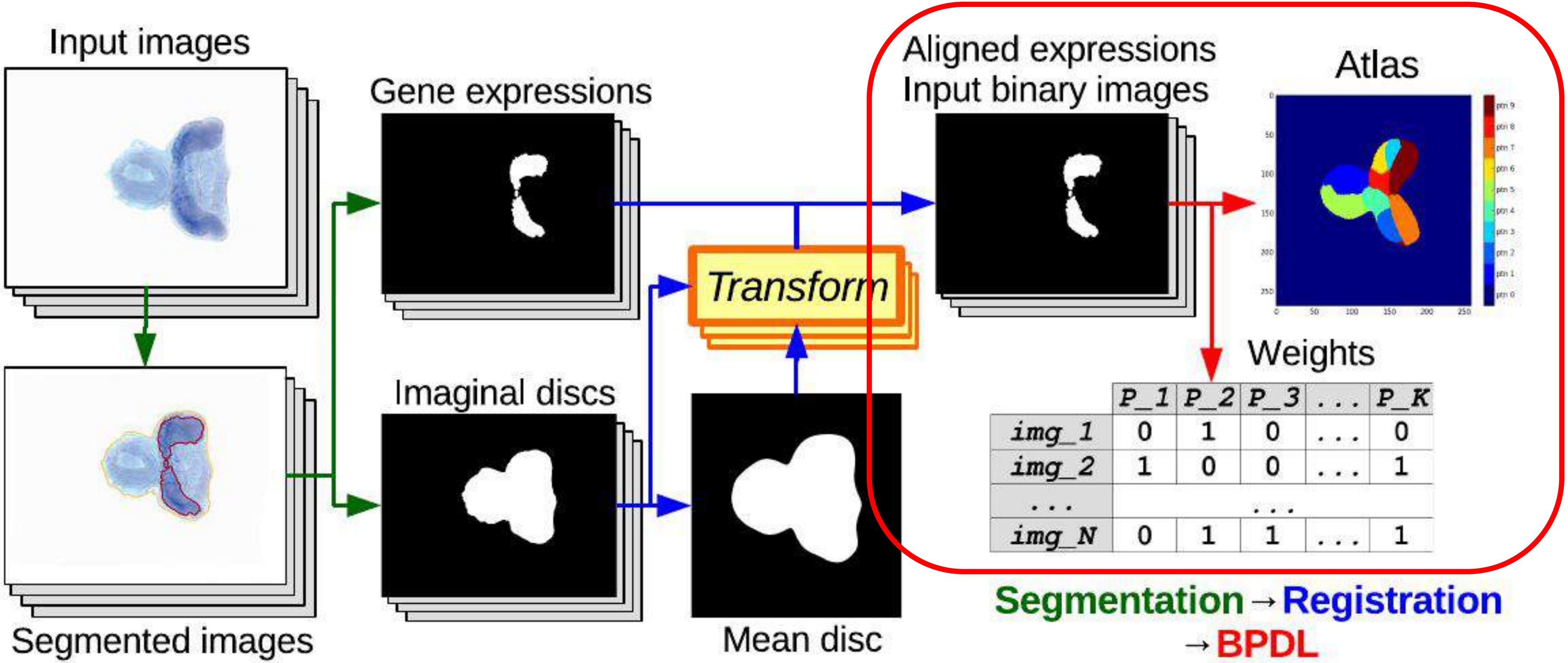
1. Problem statement
2. Used datasets
3. Instance segmentation
    a. Structure (tissue) segmentation on superpixels
    b. Center detection and ellipse fitting
    c. Region growing on superpixels
4. Binary Pattern Dictionary Learning
5. Conclusion & Future work

Binary Pattern Dictionary Learning of gene expressions



Borovec J., Kybic J. (2016) **Binary Pattern Dictionary Learning for Gene Expression Representation in Drosophila Imaginal Discs.** In: Computer Vision – ACCV 2016 Workshops. Lecture Notes in Computer Science, vol 10117, Springer, DOI: 10.1007/978-3-319-54427-4_40.

Pipeline: segmentation - registration - BPDL



Decomposition methods

Formulation

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

- Standard approaches
  - Non Negative Matrix Factorisation
  - Fast Independent Component Analysis
  - Sparse Principal Component Analyses
  - Dictionary Learning with Matching pursuit

Formulation

- Image representation

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

- Similarity measure (Hamming distance)

$$F(\mathbf{g}, \mathbf{y}, \mathbf{w}) = \sum_{i \in \Omega} [\![\mathbf{g}_i \neq \hat{\mathbf{g}}_i]\!]$$

- Regularize neighbouring pixels

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

- Optimization criterion

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

Alternating minimization

- Update weights - maximise overlap

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

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

- Update atlas

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

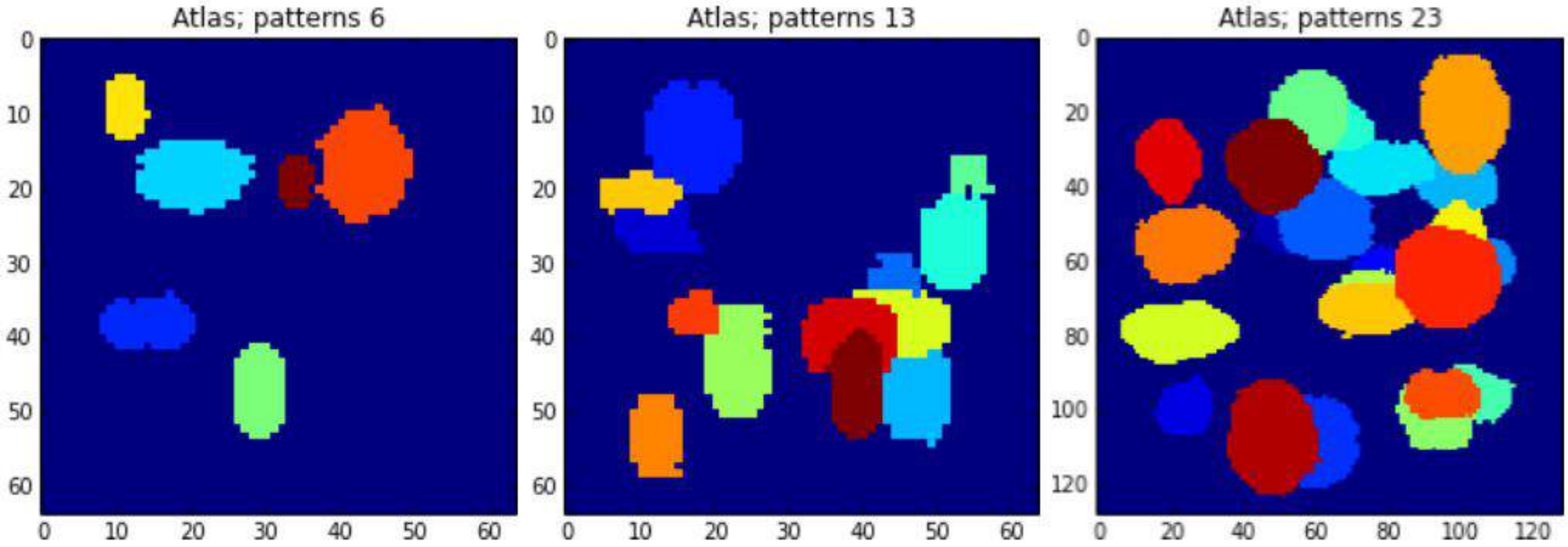
Algorithm

Algorithm 1 General schema of BPDL algorithm.

```
initialise atlas y
while not converged do
    update weights w ∈ W
    reinitialise empty patterns in y*
    update atlas y* via Graph Cut
end while
```

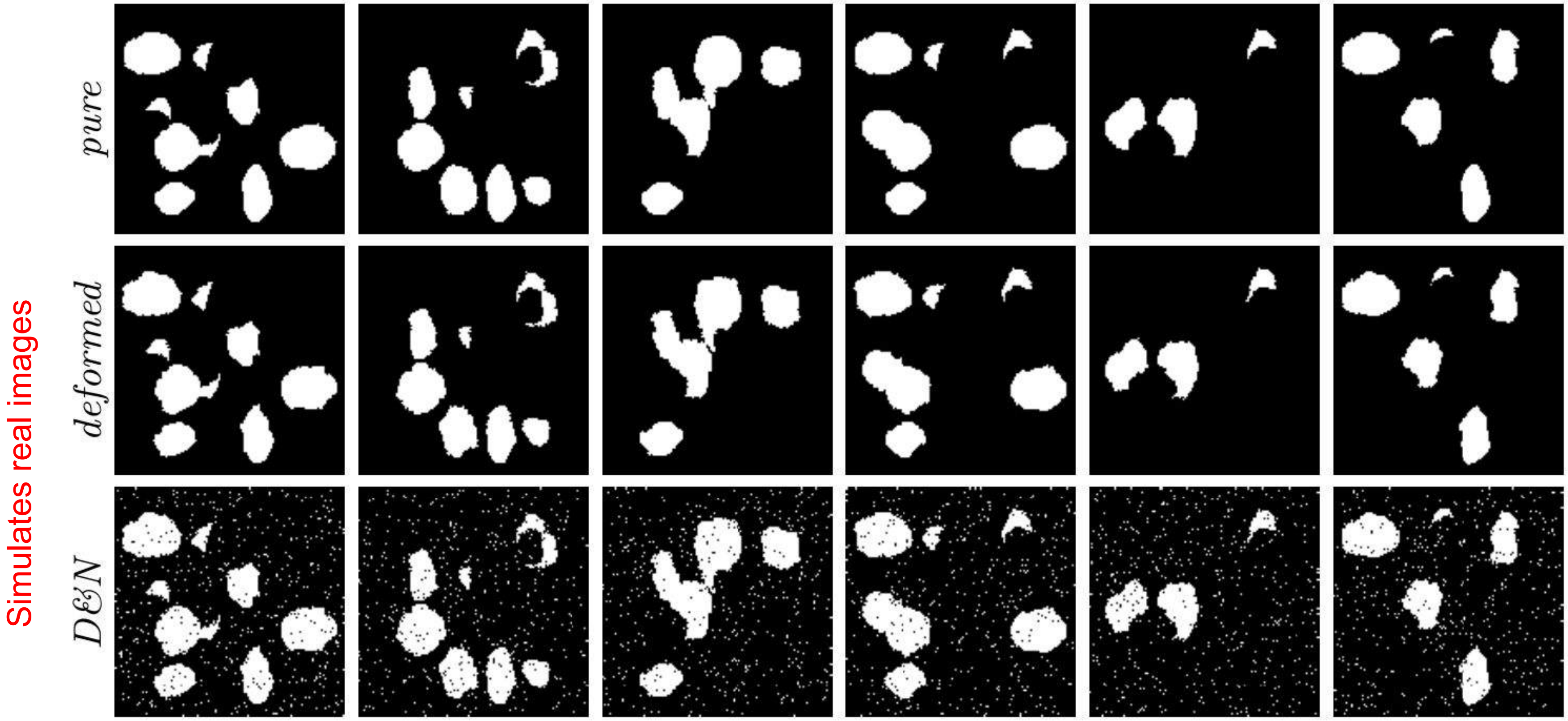
It makes the algorithm more robust to initialisation.

Synthetic datasets



v0 v1 v2

Synthetic images



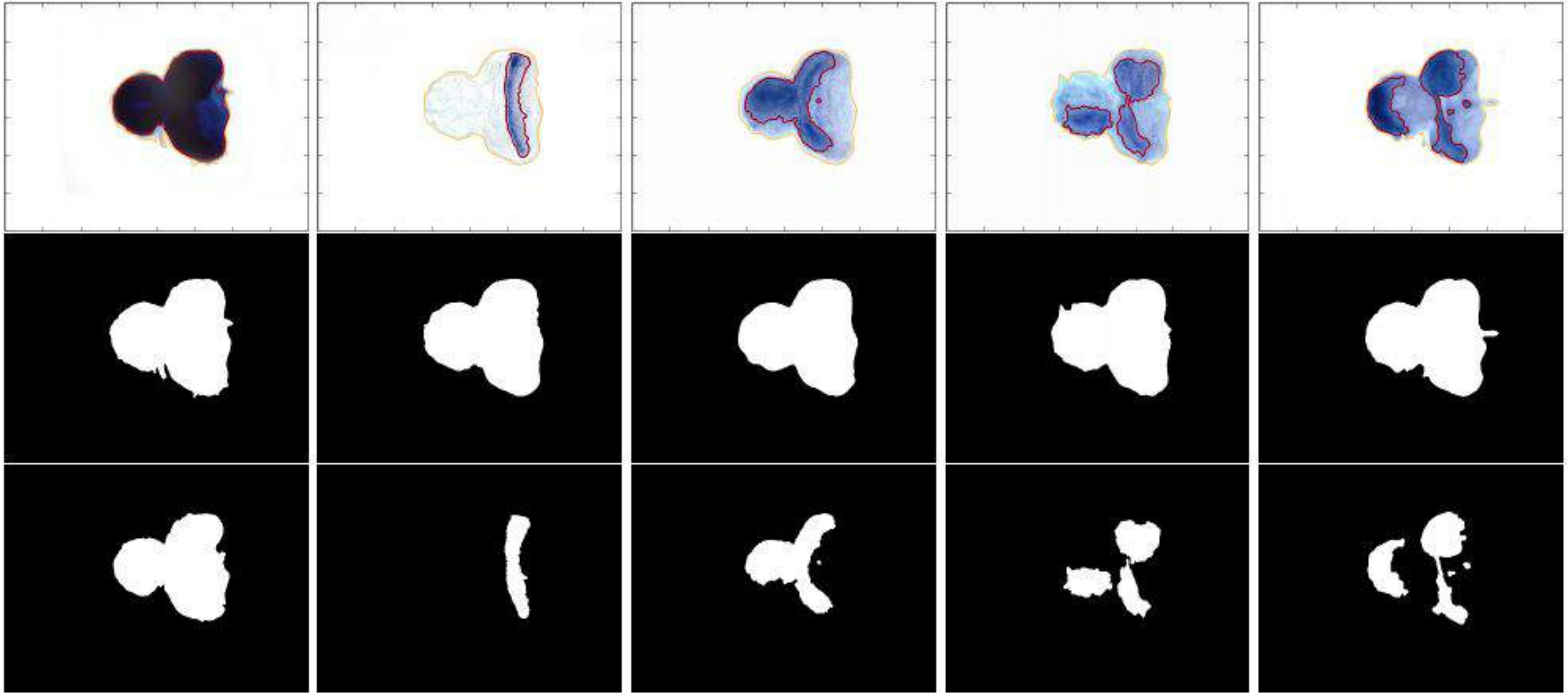
Comparison on synth. images

datasets		NMF	FastICA	sPCA	DL	BPDL
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**

Comparison on synth. images

datasets		NMF	FastICA	sPCA	DL	BPDL
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**

Input segmented imaginal discs

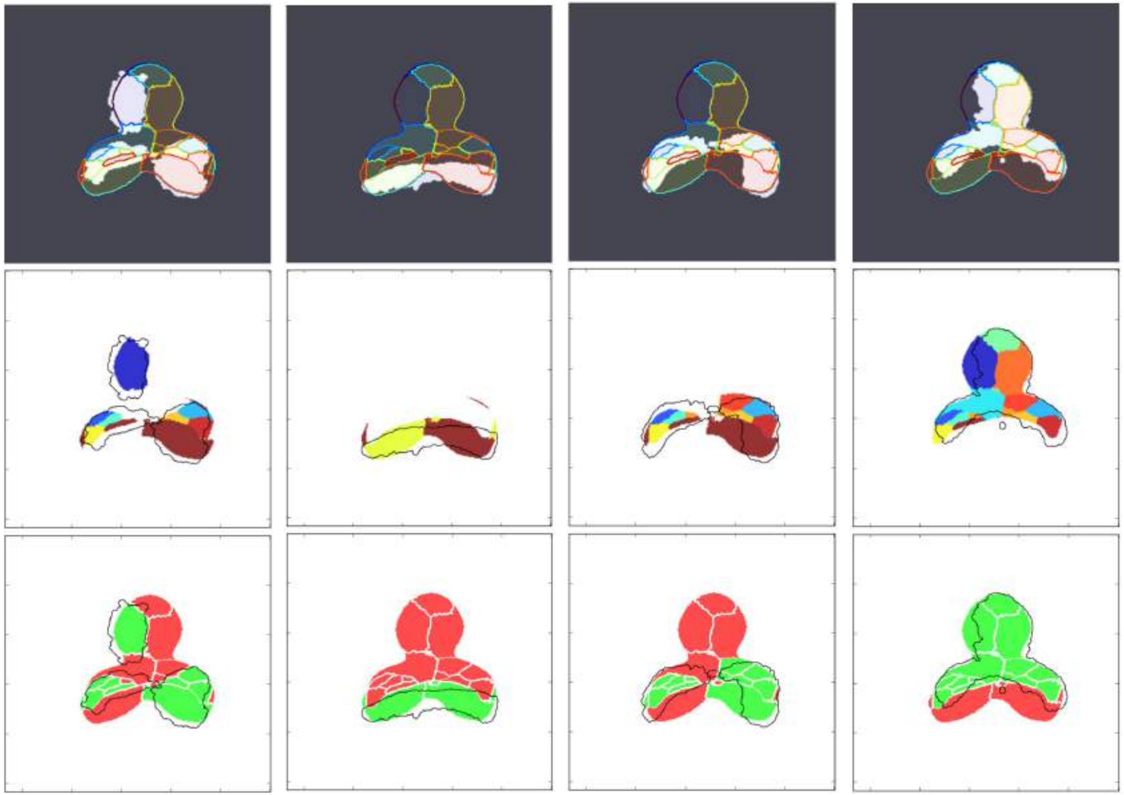


Visualise results on imaginal discs

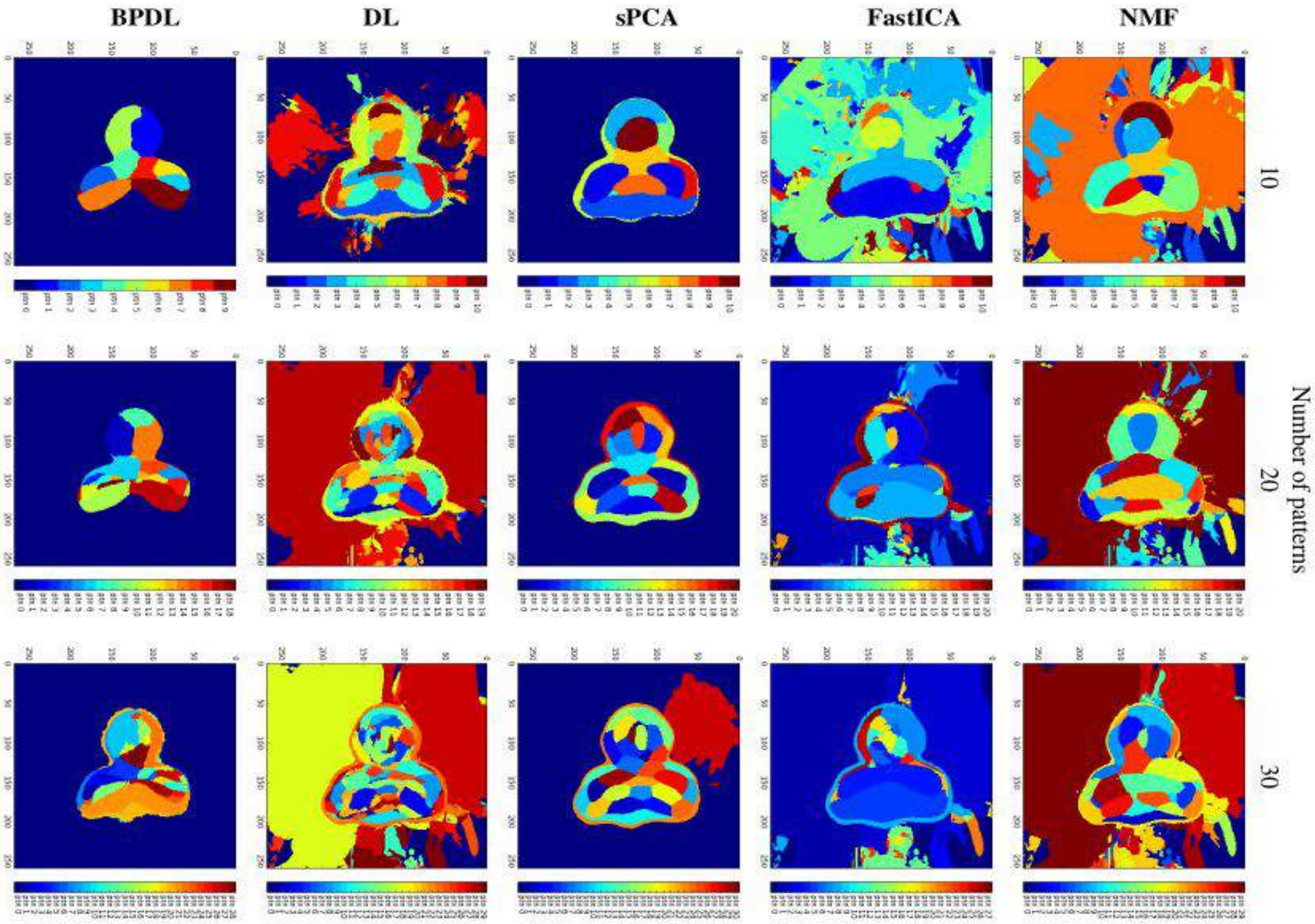
Gene & atlas

Used patterns

Pattern activation



Extracted Atlases for Imaginal discs

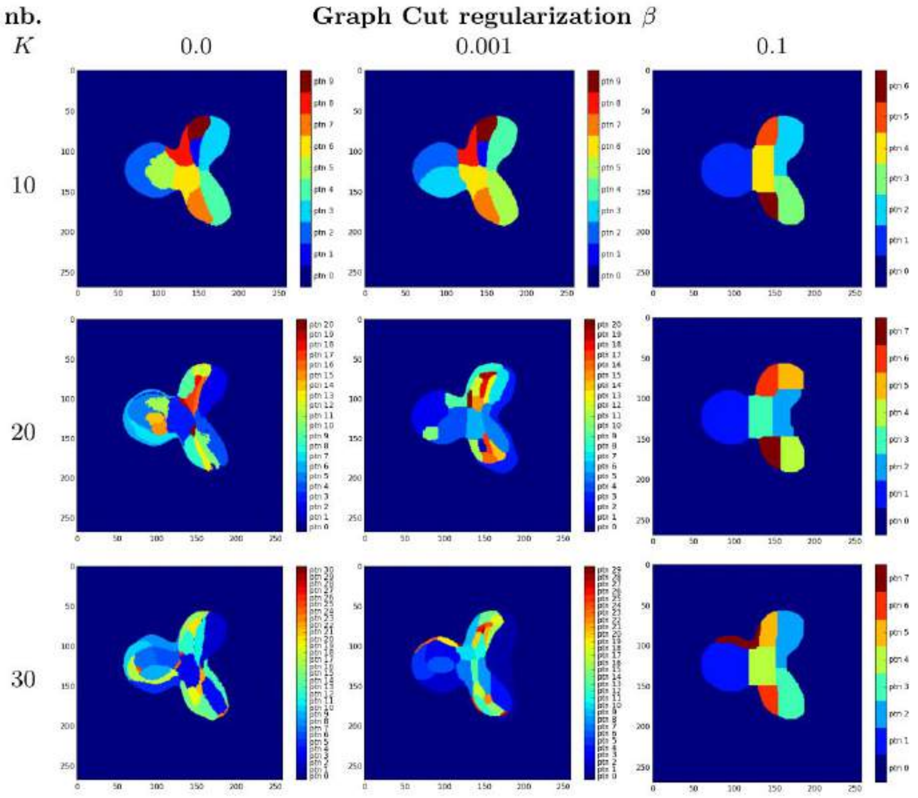


Results on imaginal discs

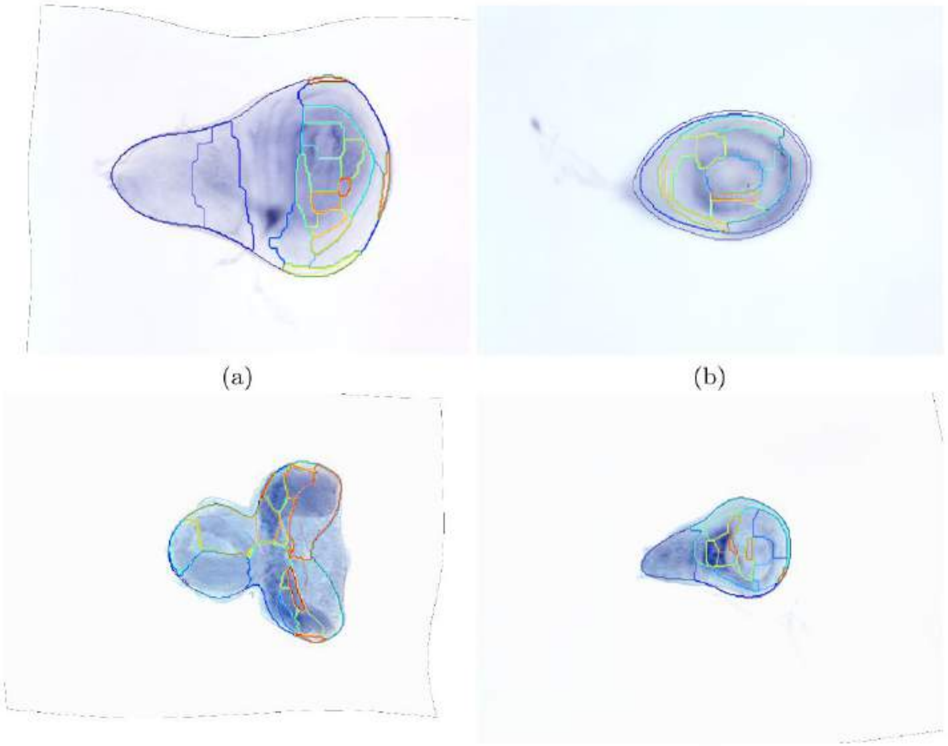
No ground truth atlases…

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

Parameter selection



& More discs



(a) (b) (c) (d)

Overview

1. Problem statement
2. Used datasets
3. Instance segmentation
    a. Structure (tissue) segmentation on superpixels
    b. Center detection and ellipse fitting
    c. Region growing on superpixels
4. Binary Pattern Dictionary Learning
5. Conclusion & Future work

Conclusion

- Presented four image processing methods:
  - Image segmentation on superpixels
  - Center detection on segmented images
  - Region growing with shape prior
  - Binary Pattern Dictionary Learning
  - **Implementation:**
    - http://borda.github.com/pyImSegm
    - http://borda.github.com/pyBPDL
- Future work
  - Complete image analysis pipeline in 2.5D
  - Try instance segmentation with CNNs
  - ...