


# Analysis of microscopy images

Jiri Borovec  
2018 @ CMP, FEE, CTU in Prague



<http://cmp.felk.cvut.cz/~borovji3/>  
<https://www.researchgate.net/publication/323120618>

# Overview

1. Brief introduction
2. Structure (tissue) segmentation on superpixels
3. Center detection and ellipse fitting
4. Region growing on superpixels
5. Binary pattern dictionary learning
6. Conclusion & Future work

# Introduction

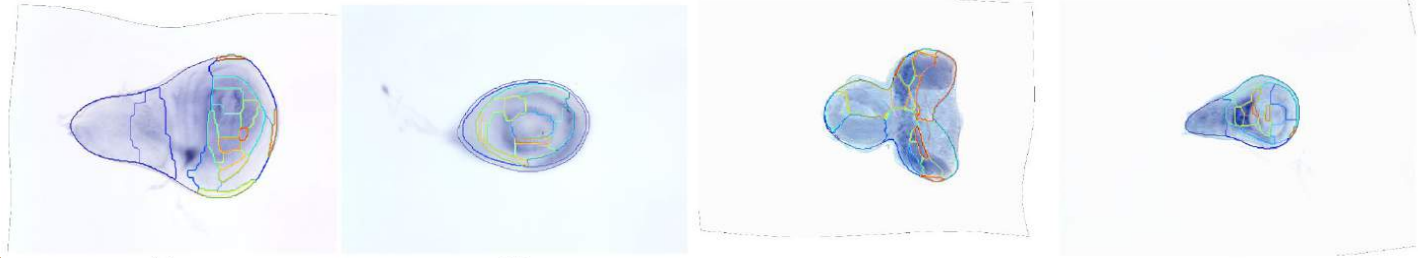
**Task:** Study of gene expressions during early development  
(ovaries & imaginal discs in larvar transition)

Study subject - *Drosophila*

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

Automatic image processing:

- Analysis requires thousands of images to be processed
- Very time consuming for experts

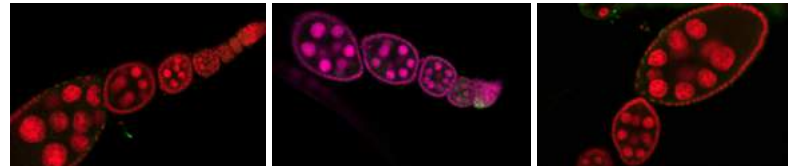
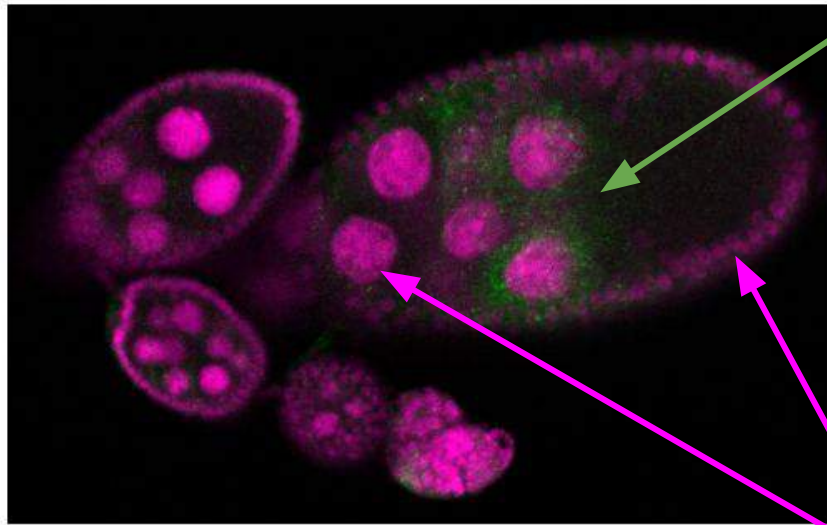


# Sample images

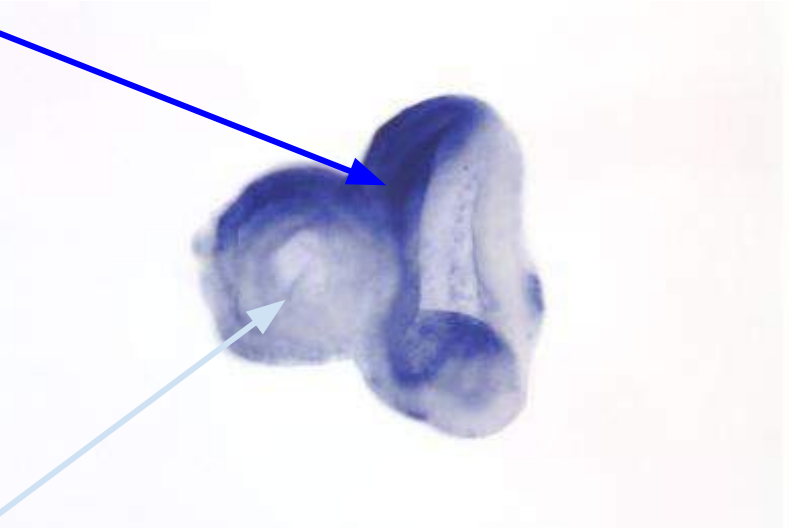
*Drosophila ovaries*

Gene activations

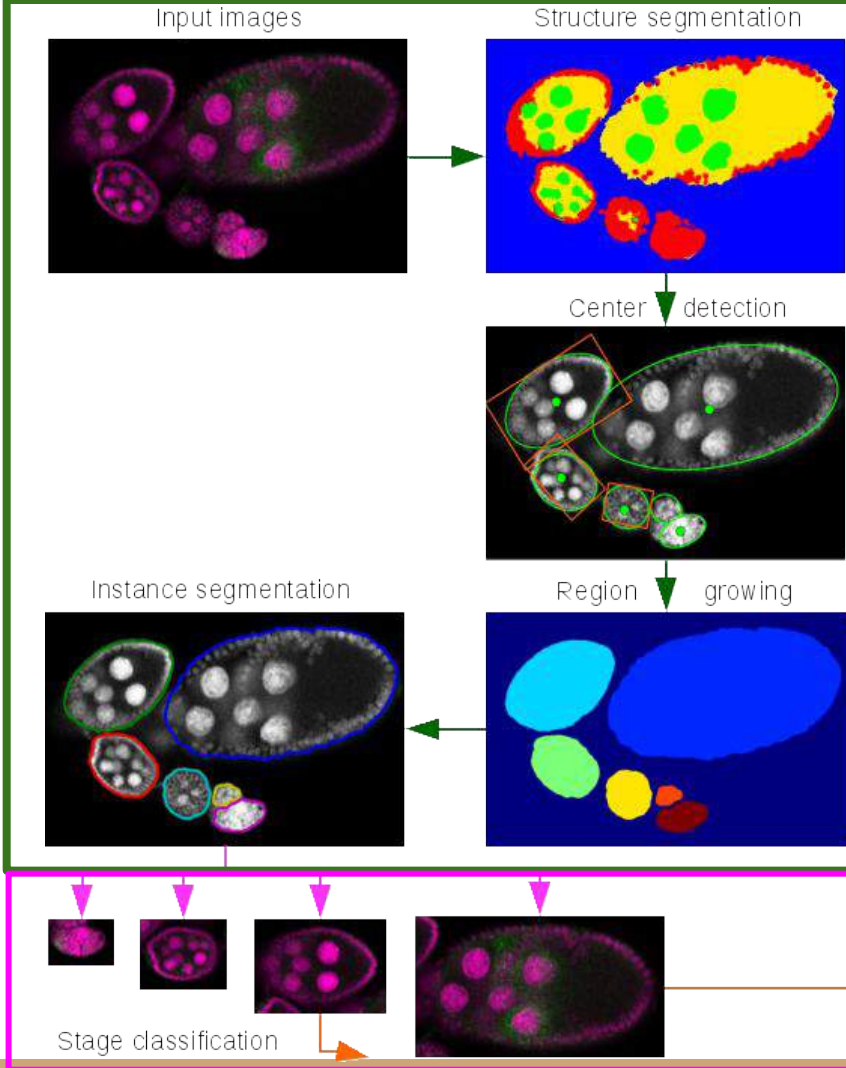
*Imaginal disc*



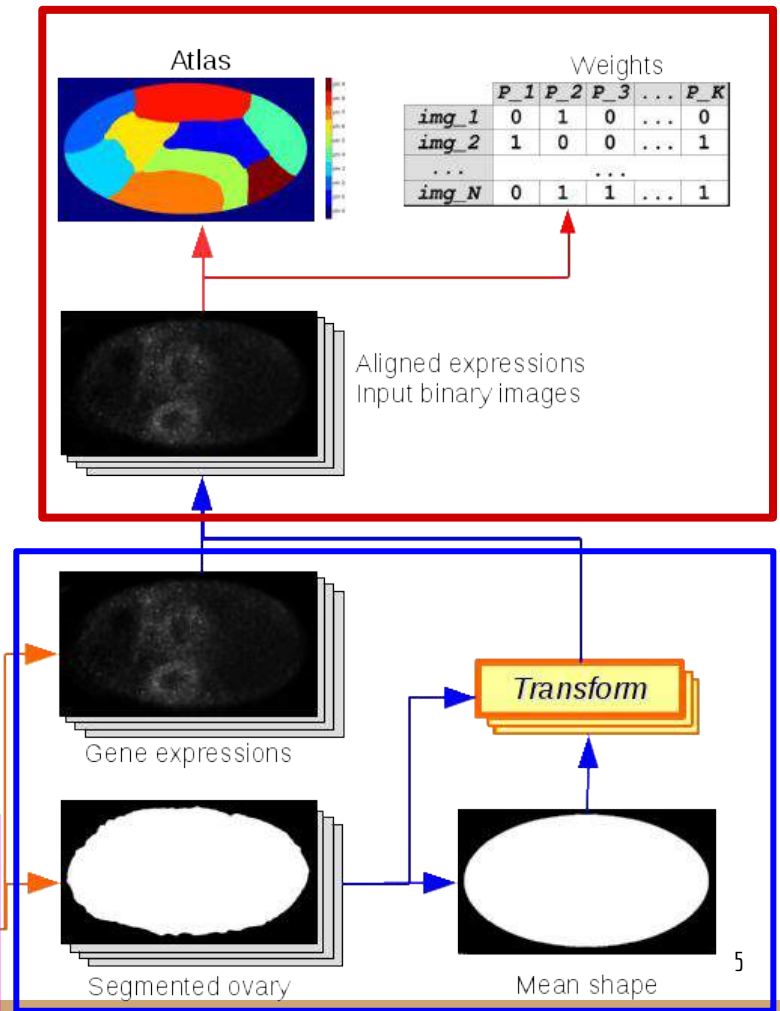
Tissue



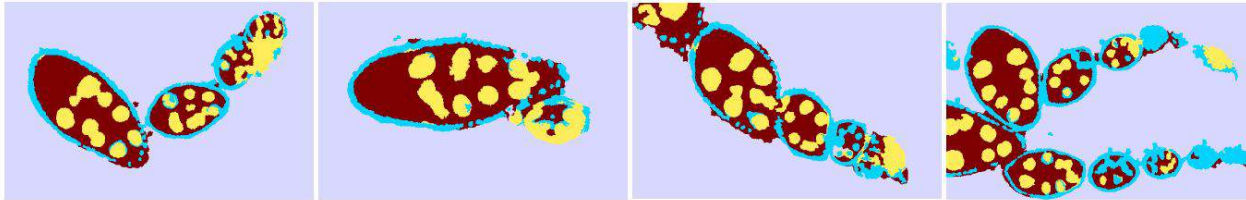
# Image analysis pipeline



## Segmentation → Classification → Registration → BPDFL



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



Borovec, J., Svihlik, J., Kybic, J., & Habart, D. (2017). **Supervised and unsupervised segmentation using superpixels, model estimation, and Graph Cut.** Journal of Electronic Imaging, 26(6), 26-26-17.

<https://doi.org/10.1117/1.JEI.26.6.061610>

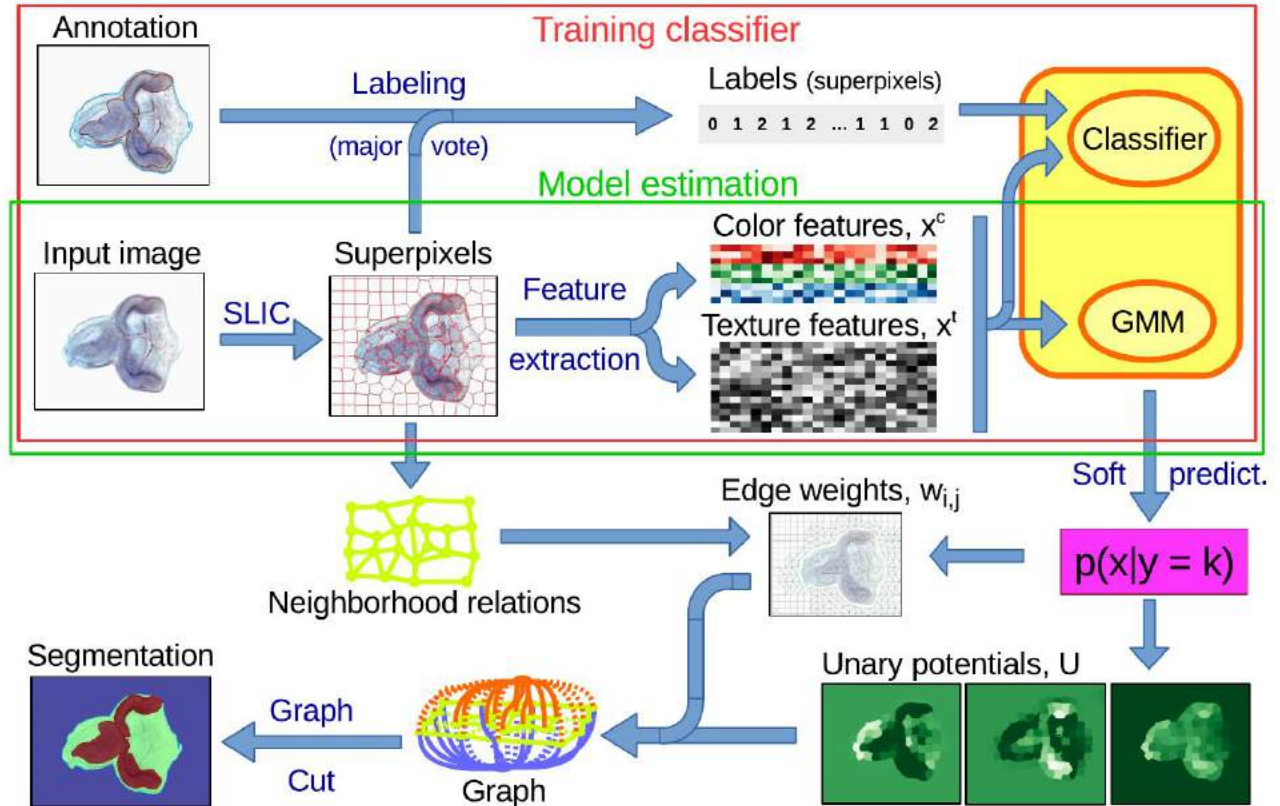
# Formulations & work-flow

Standard formulation

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

Image segmentation Y

1. Superpixels
2. Feature extraction X
3. Class modelling
  - a. Training classif.
  - b. Model estetim.
4. Graph-Cut regularization



# Segmentation quality comparison

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

Used metric: **F<sub>1</sub> score**

Standard methods:

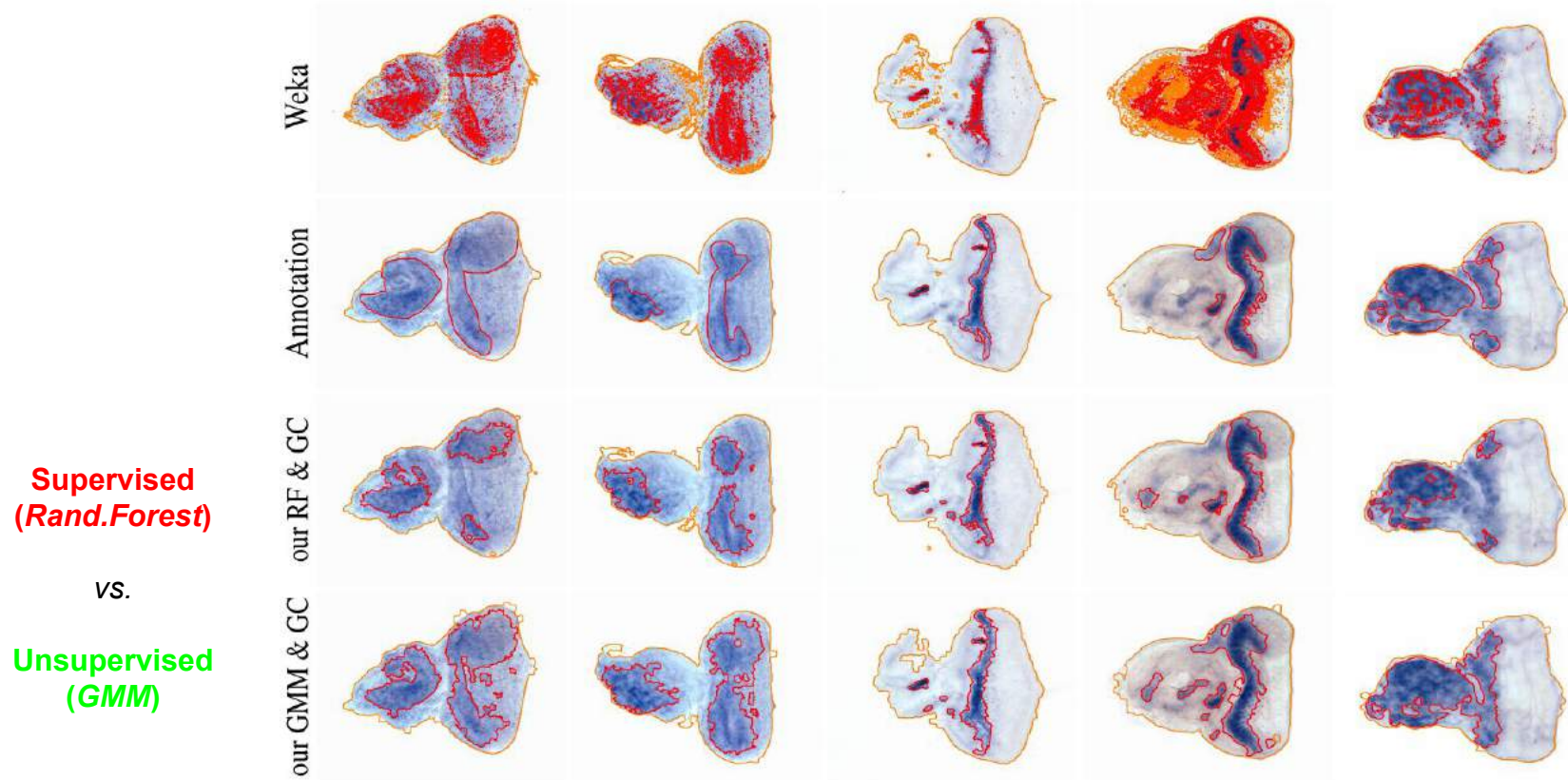
- Weka (ImageJ) uses color & texture features and RF classifier and GC(smoot. cost, edge cost)
- Supertextons uses texture features and kNN classifier
- $Y_A$  ideal result on superpixels (major vote per superpixel)

**Shortcuts:**

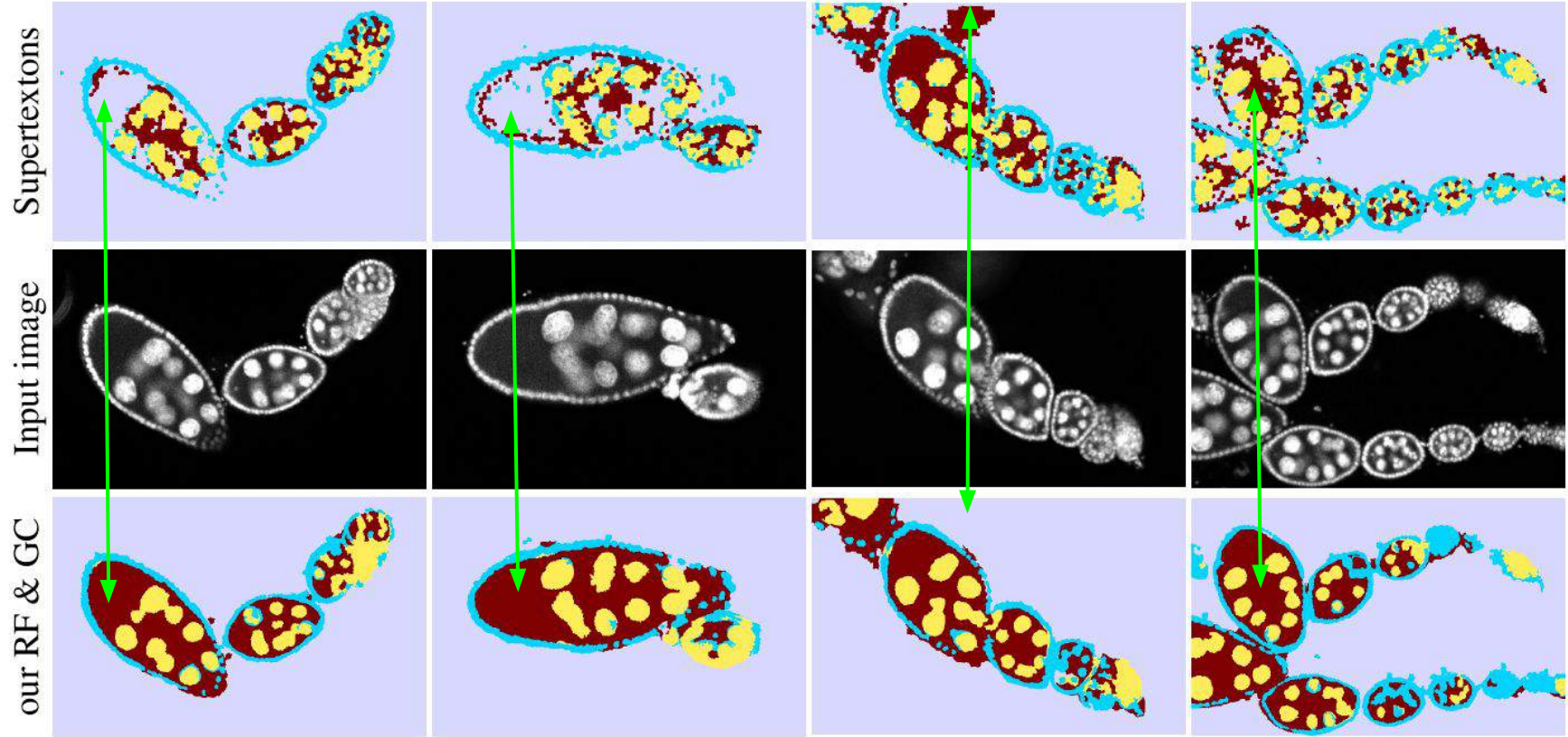
- GC - Graph-Cut
- RF - Random Forest
- GMM - Gaussian Mixture Model
- gr. - learned over whole set of images



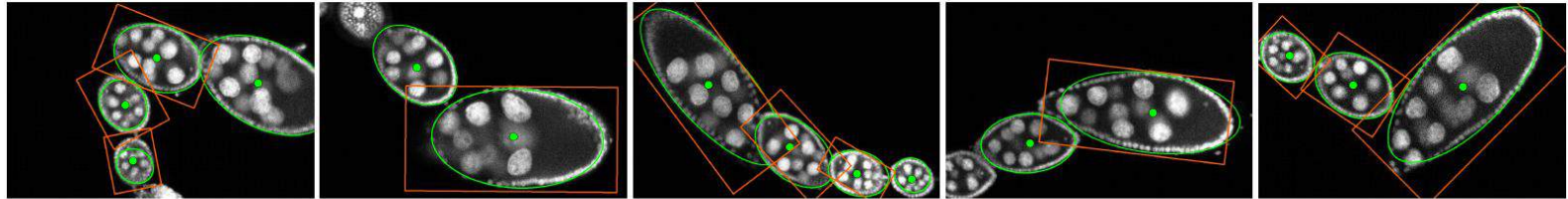
# Supervised vs unsupervised segmentation



# Advantage of using Graph Cut regularization



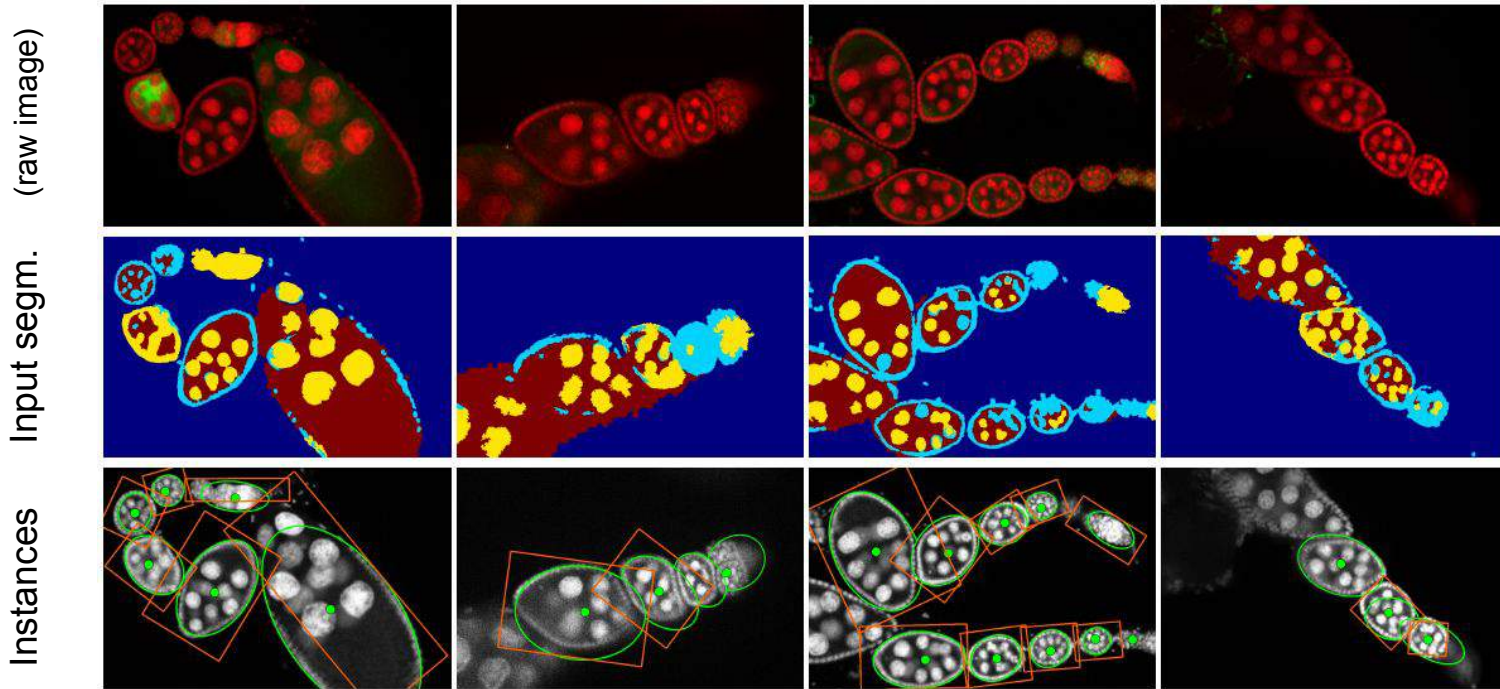
# *Detection and localization of Drosophila egg chambers in microscopy images*



Borovec, J., Kybic, J., & Nava, R. (2017). **Detection and Localization of Drosophila Egg Chambers in Microscopy Images.** In Q. Wang, Y. Shi, H.-I. Suk, & K. Suzuki (Eds.), 8th International Workshop Machine Learning in Medical Imaging (MLMI) (pp. 19–26). Cham: Springer International Publishing. [https://doi.org/10.1007/978-3-319-67389-9\\_3](https://doi.org/10.1007/978-3-319-67389-9_3)

# Center detections & ellipse approximations

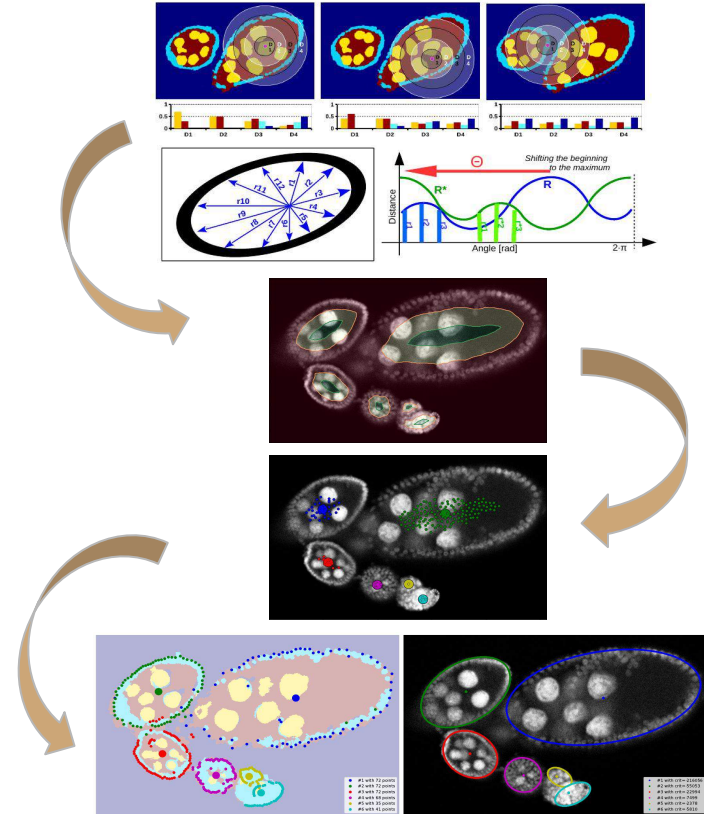
From input semantic segmentation we detect individual egg instances





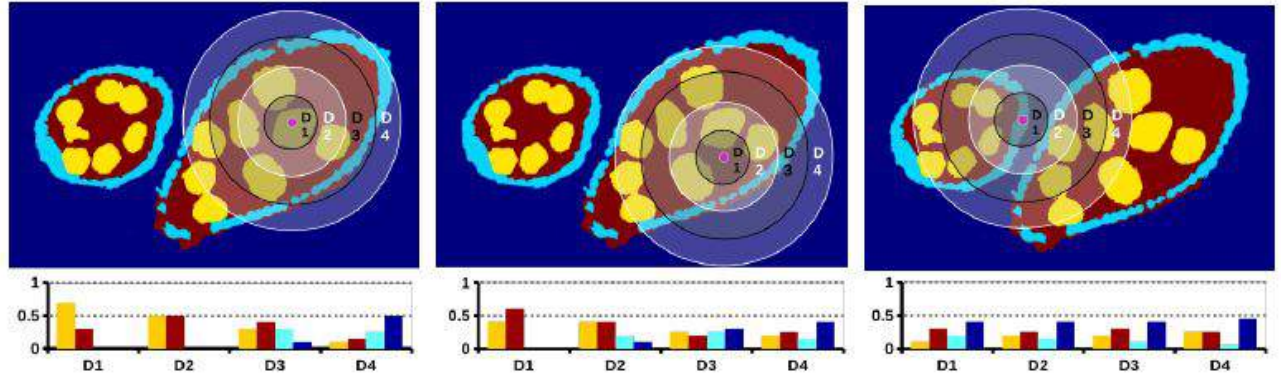
# Schema & work-flow

1. Extract pixel features
2. Train center detection classifier
3. Group center candidates
4. Ellipse fitting

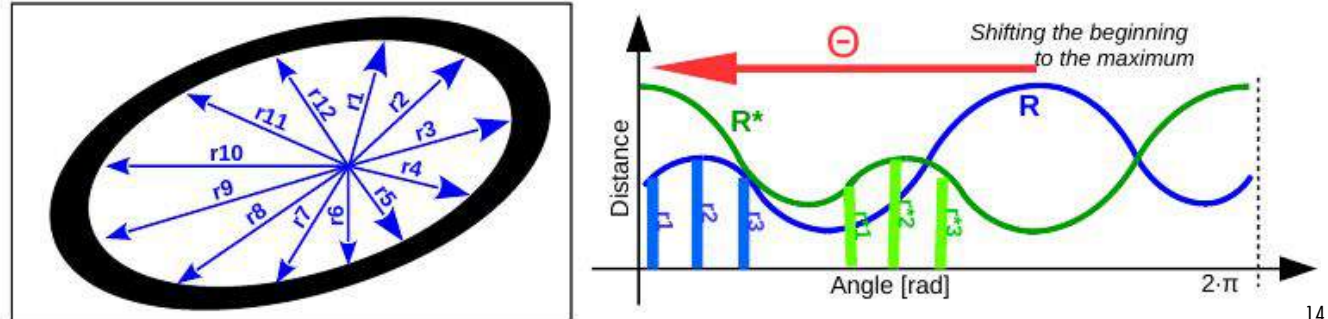


# Features for center detection

- Label histogram



- Ray features



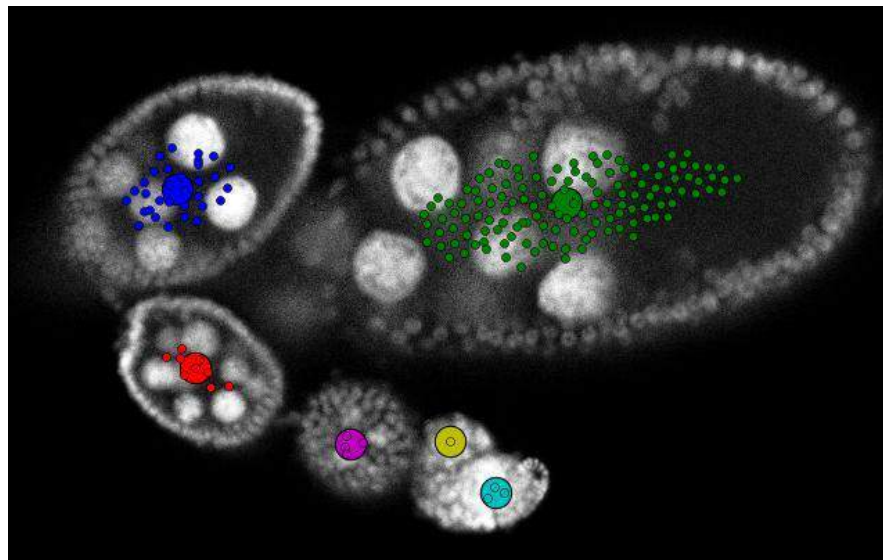
# Classification & Grouping

Train classifier on 3 classes:  
(discrete zones for center appearances)

**Positive** - **Neutral** - **Negative**

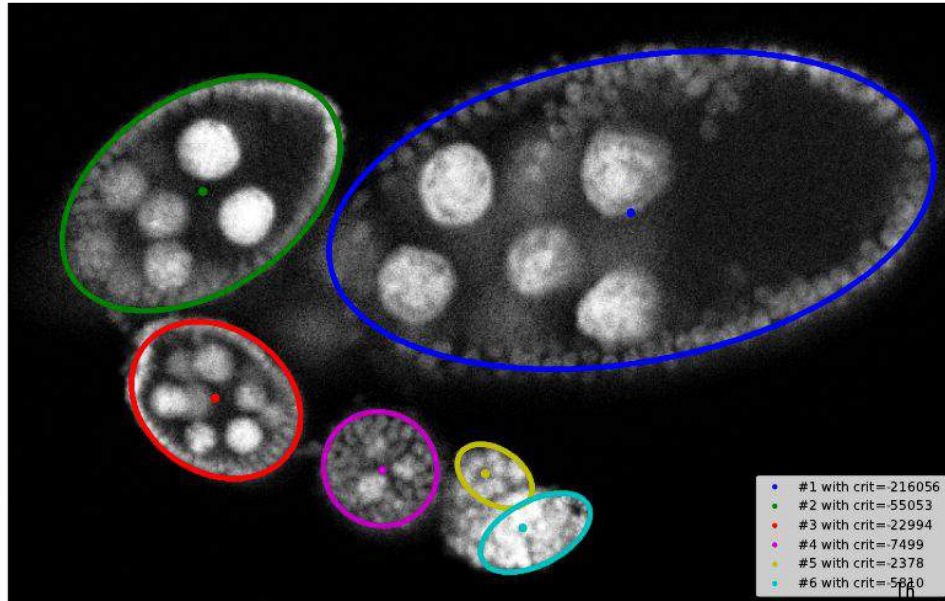
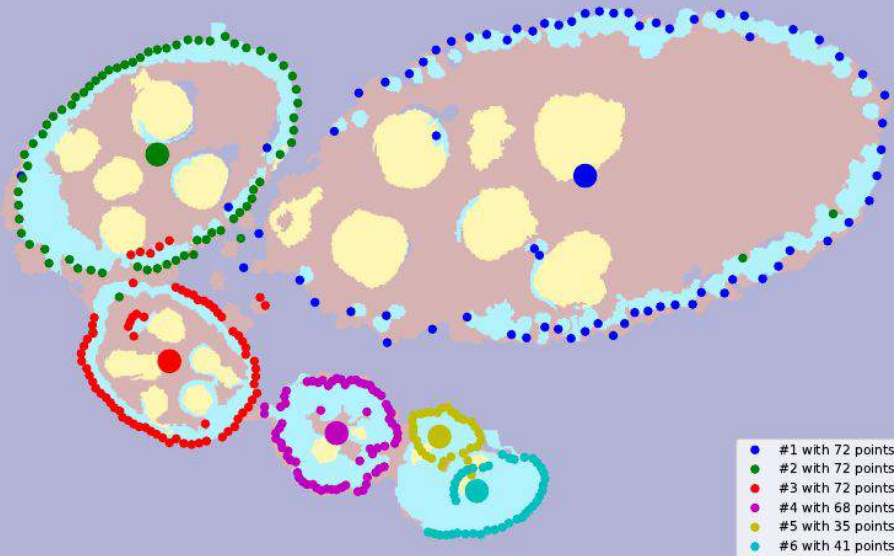


Clustering of center candidates with [DBSCAN](#)  
(used max distance as 3x superpixel size)



# Ellipse fitting

- RANSAC-like strategy on boundary points
- maximise foreground labels inside estimated ellipses

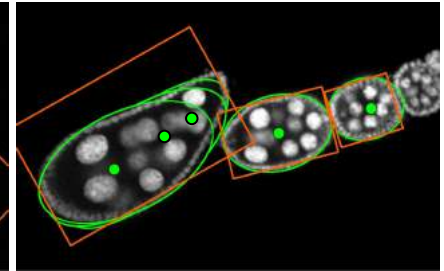
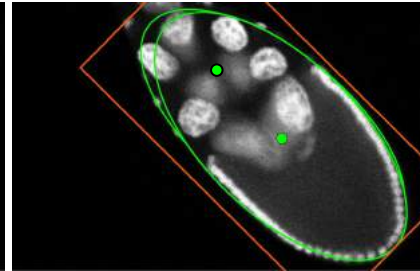
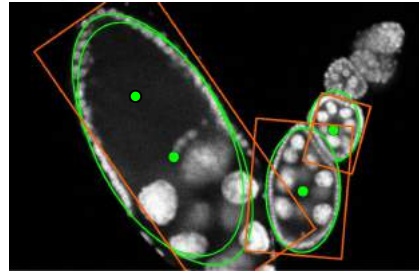
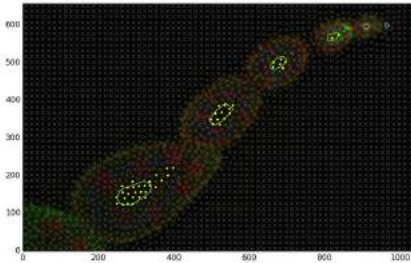


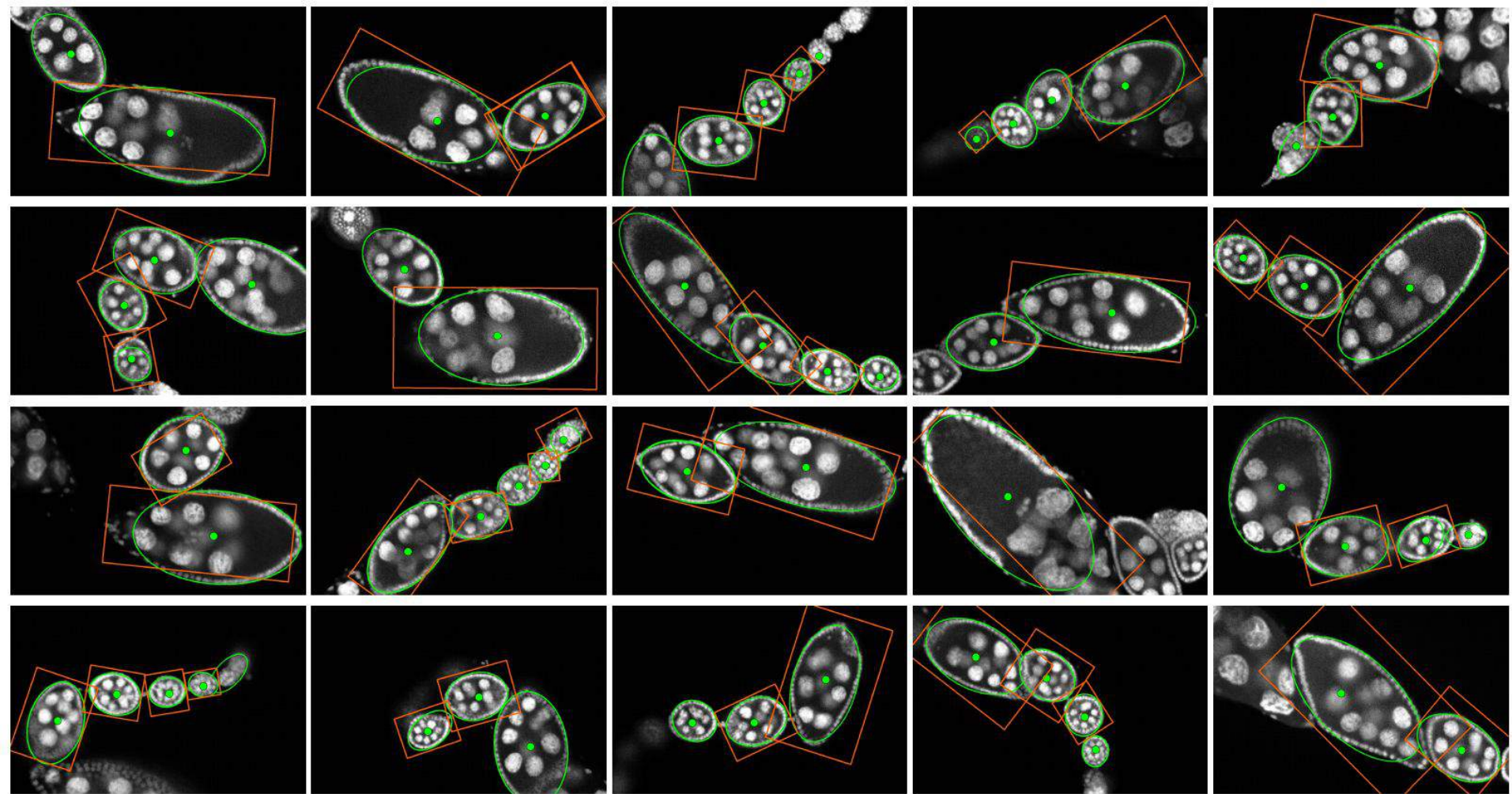


# Ellipse pruning & detection results

Group center detections using ellipse fitting

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%)				

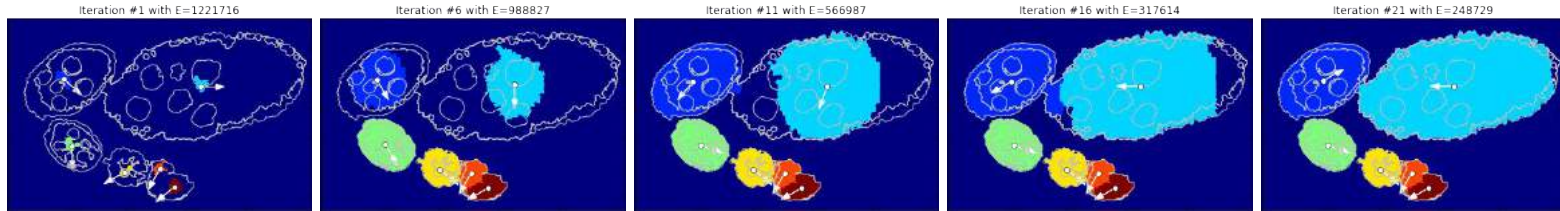




**Red rectangle** - expert annotation

**Green dot/ellipse** - predict center detection & ellipse fitting

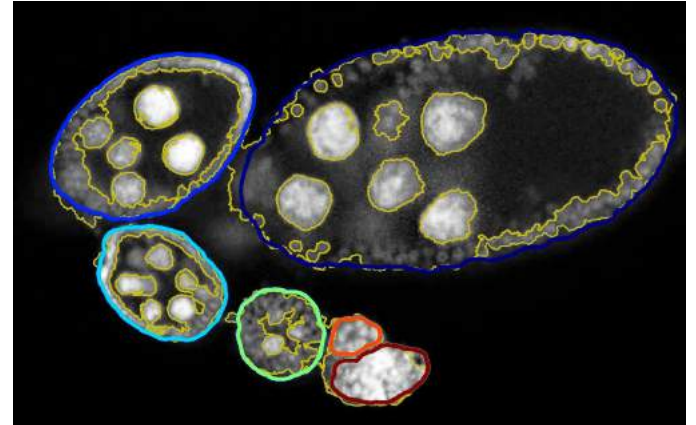
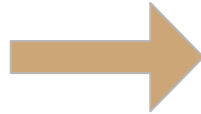
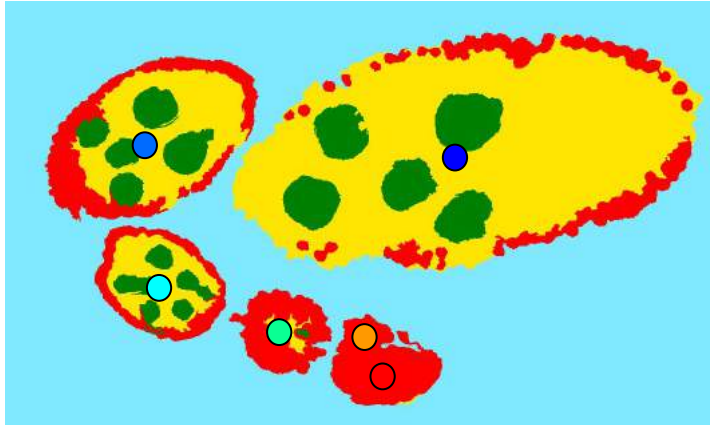
# *Region growing using superpixels with learned shape prior*



Borovec, J., Kybic, J., & Sugimoto, A. (2017). **Region growing using superpixels with learned shape prior**. Journal of Electronic Imaging, 26(6), 26-26-14. <https://doi.org/10.1117/1.JEI.26.6.061611>

# Formulation

- Grow initial seeds
- Use semantic segmentation
- Superpixels for speedup
- Enforce compact objects (eggs)
- Use learned shape prior



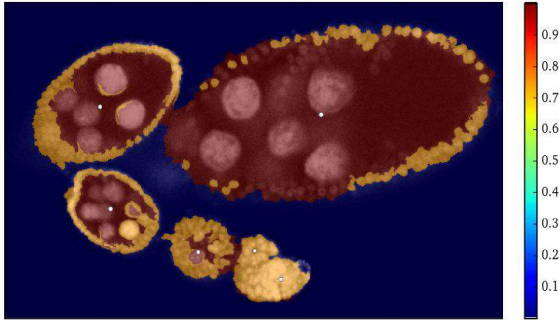


# Region growing - variational framework

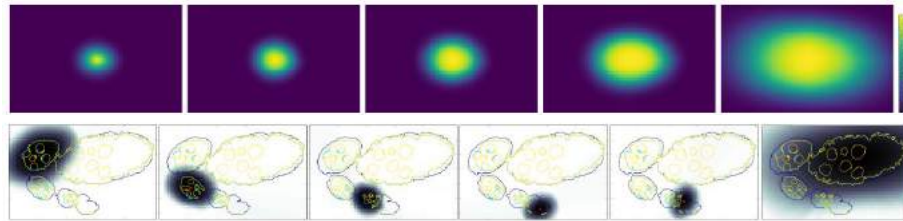
Formulated as:

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

Appearance model



Shape prior & Mixture model



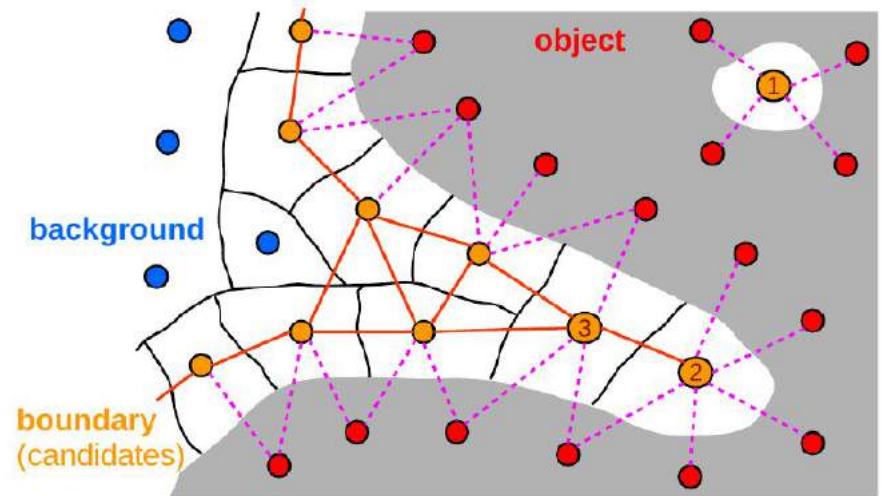
Pairwise regularization

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

Iterative adding superpixels on object boundary.

# Region growing - optimisation

- Iterative growing on object boundaries
- Alternating: region growing & shape model parameter update
- Strategies:
  - Greedy growing
  - Graph Cut on boundaries
    - Binary (for each object independently)
    - Multi-class (for all objects at once)
  - Object swapping (labels between objects)

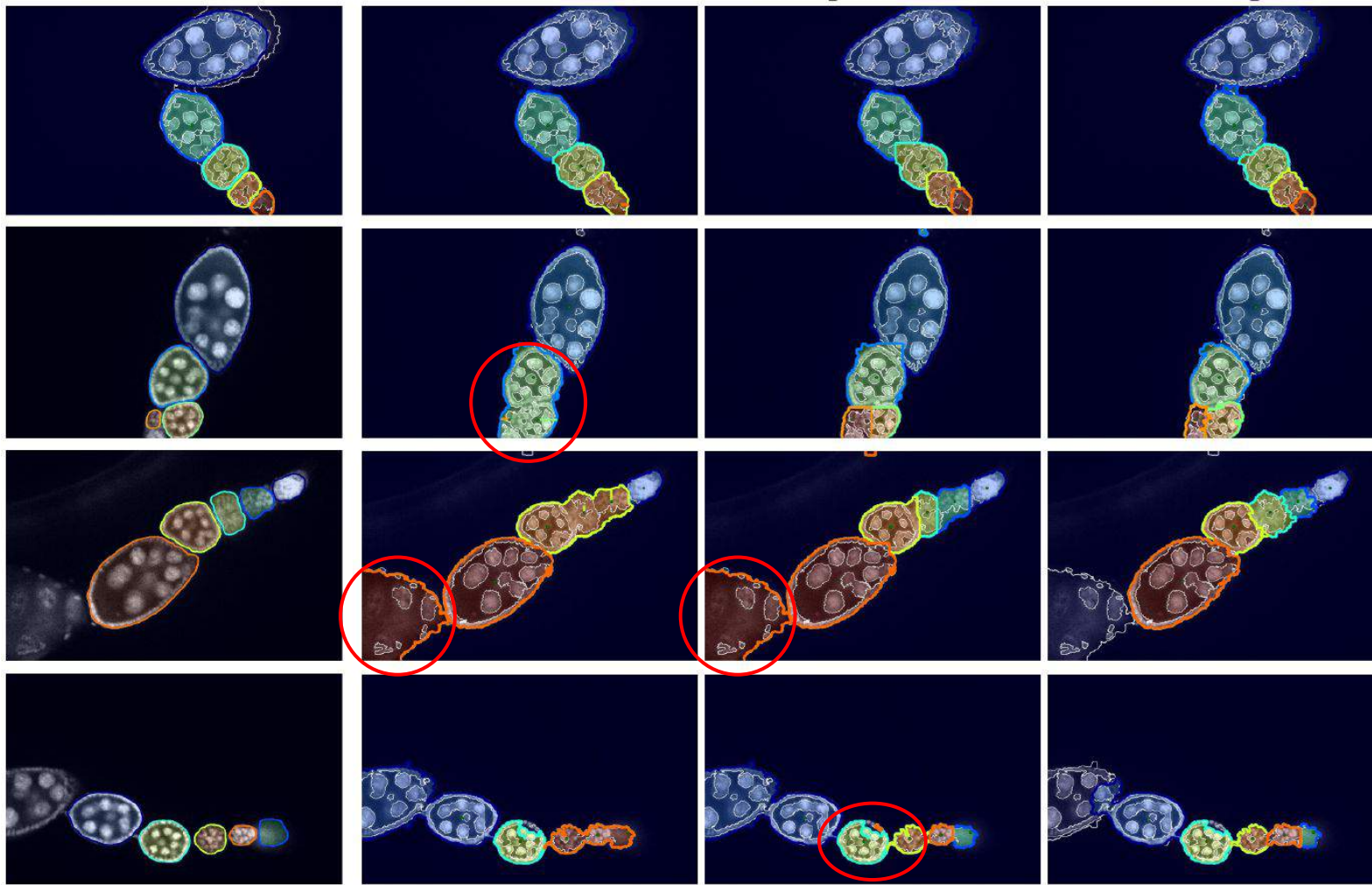


Annotation

Watershed

GC (pixel-wise)

RG2Sp



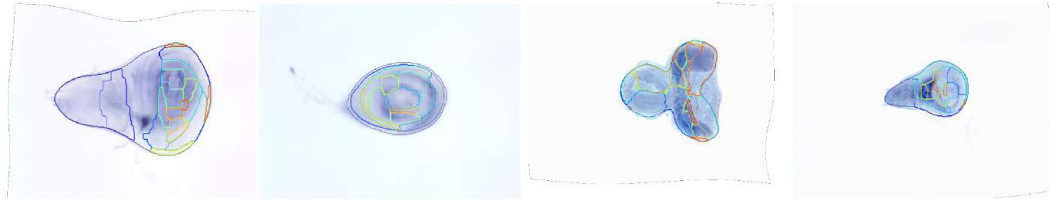
# Segmentation quality comparison

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



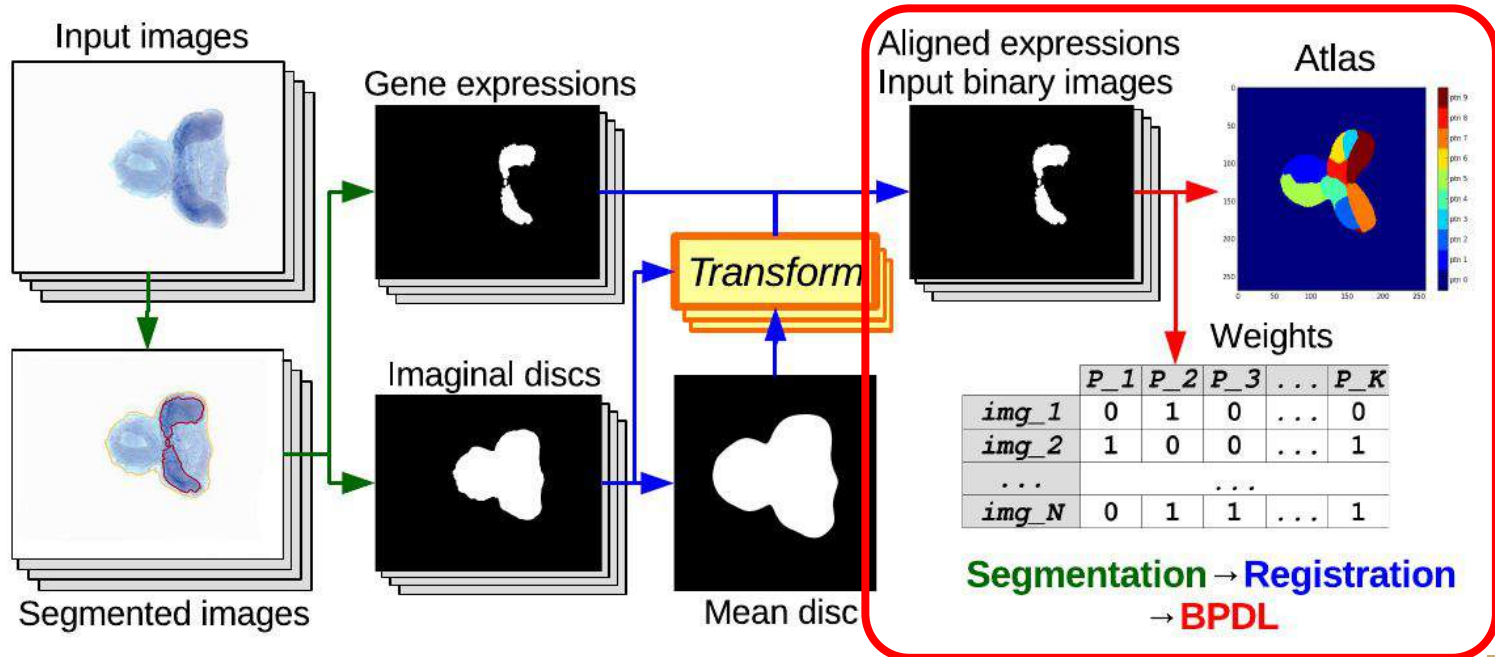
# *Binary Pattern Dictionary Learning for gene expressions*



Borovec, J., & Kybic, J. (2016). **Binary pattern dictionary learning for gene expression representation in drosophila imaginal discs.** In *Mathematical and Computational Methods in Biomedical Imaging and Image Analysis (MCBMIIA), ACCV workshops* (pp. 555–569). Springer. [https://doi.org/10.1007/978-3-319-54427-4\\_40](https://doi.org/10.1007/978-3-319-54427-4_40)

# BPDL with preprocessing (segm. & regist.)

Estimate an atlas of non-overlapping spatial patterns + binary encoding for each image. Best approximation of the input binary images.



# Formulation

- Minimizing binary differences between input  $g$  and reconstructed  $\hat{g}$  images. atlas  $y$ , active patterns  $w$

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

- Similarity measure (Hamming dist.)      reconstructed image

$$F(\mathbf{g}, \mathbf{y}, \mathbf{w}) = \sum_{i \in \Omega} \llbracket \mathbf{g}_i \neq \hat{\mathbf{g}}_i \rrbracket \quad \hat{\mathbf{g}} = \sum_{l \in \mathbf{L}} \mathbf{w}_l \cdot \llbracket \mathbf{y} = l \rrbracket$$

- Spatial regularisation

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


# Alternating minimization

- Update weights

$$w_l = \llbracket P(\mathbf{g}, \mathbf{y}, l) \geq \sigma \rrbracket \quad \text{where } \sigma = 1$$

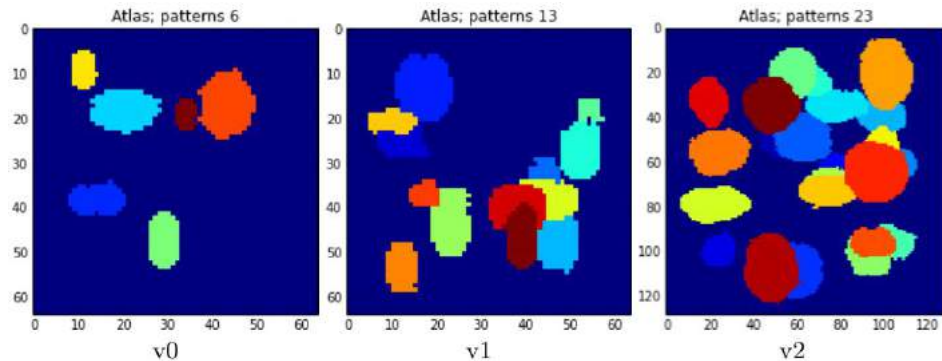
$$\text{and } P(\mathbf{g}, \mathbf{y}, l) = \frac{\sum_{i \in \Omega, \mathbf{y}_i = l} \llbracket \mathbf{g}_i = 1 \rrbracket}{\sum_{i \in \Omega, \mathbf{y}_i = l} \llbracket \mathbf{g}_i \neq 1 \rrbracket} = \frac{\|\llbracket \mathbf{y} = l \rrbracket\|}{\sum_{i \in \Omega, \mathbf{y}_i = l} (1 - \mathbf{g}_i)} - 1$$

- Update atlas by minimising

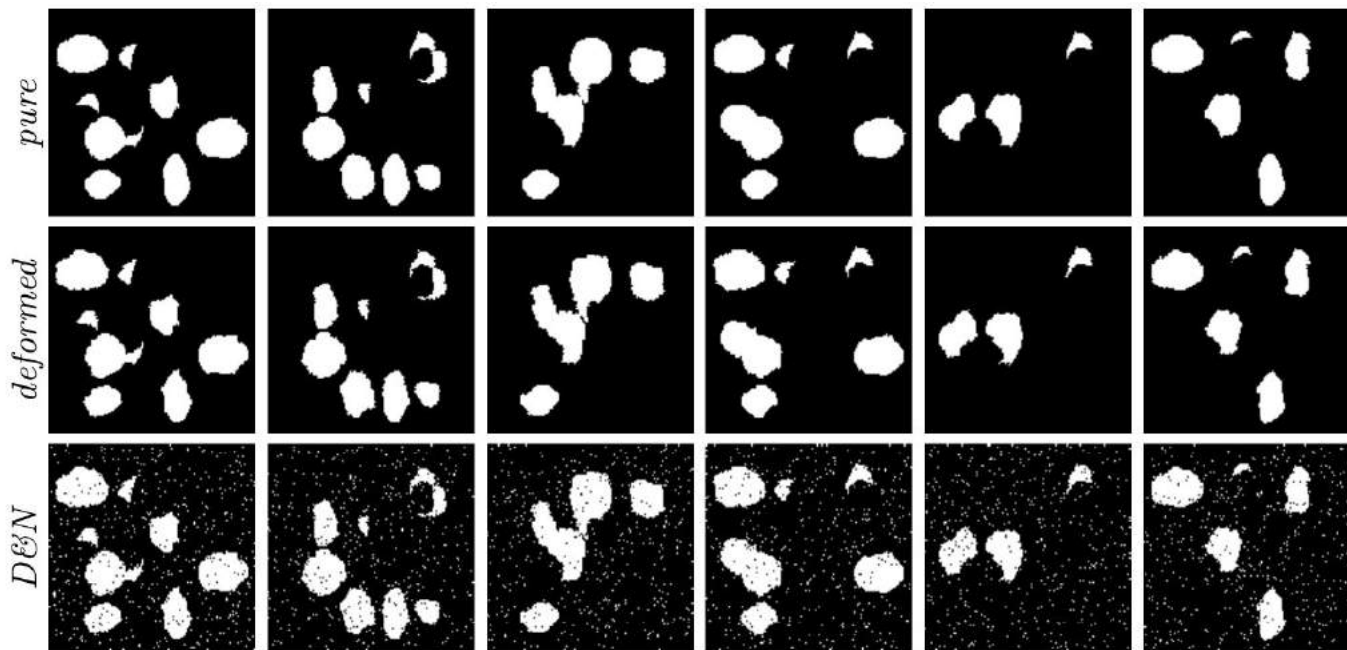

$$\frac{1}{N} \sum_{i \in \Omega} \sum_n \underbrace{\left| \mathbf{g}_i^s - \sum_{l \in \mathbb{L}} \mathbf{w}_l^s \cdot \llbracket \mathbf{y} = l \rrbracket \right|}_{U_i(\mathbf{y}_i)} + \sum_{\substack{i, j \in \Omega, i \neq j, \\ d(i, j) = 1}} \llbracket \mathbf{y}_i \neq \mathbf{y}_j \rrbracket$$

- Reinitialise unused pattern labels

# Results - synthetic data



Simulates real images



# Comparison on synthetic data

datasets		NMF	FastICA	sPCA	DL	BPDL
v1		<i>(size 64 × 64 px, 13 patterns)</i>				
<i>pure</i>	ARS	<b>1.0</b>	<b>1.0</b>	0.992	<b>0.995</b>	<b>0.999</b>
	diff.	<b>0.0</b>	<b>0.0</b>	0.0298	<b>0.019</b>	<b>0.0</b>
	time	<b>2.333</b>	340.32	18.291	737.47	<b>6.029</b>
<i>deform</i>	ARS	0.785	<b>0.948</b>	0.780	0.779	<b>0.992</b>
	diff.	0.017	<b>0.004</b>	0.029	0.033	<b>0.005</b>
	time [s]	<b>4.001</b>	312.18	15.000	700.03	<b>7.561</b>
<i>D&amp;N</i>	ARS	0.091	<b>0.878</b>	0.009	0.0727	<b>0.951</b>
	diff.	0.048	<b>0.010</b>	0.061	0.0499	<b>0.003</b>
	time [s]	<b>4.490</b>	439.04	11.420	697.599	<b>9.562</b>

**NMF** ... Non-negative Matrix Factorization

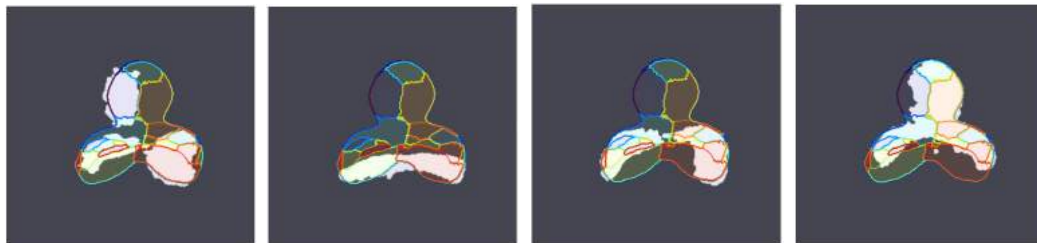
**DL** ... Dictionary Learning with Matching pursuit

**FastICA** ... Fast Independent Component Analysis

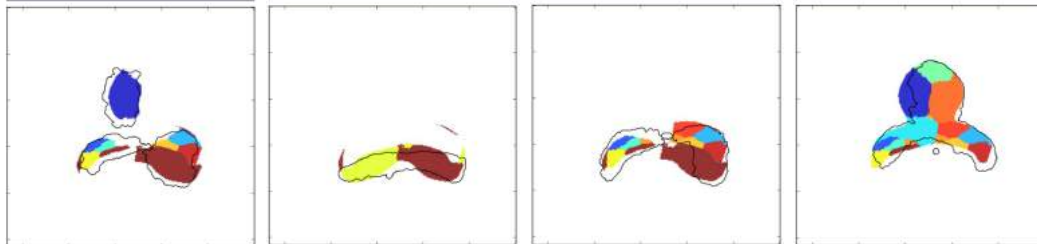
**sPCA** ... Sparse Principal Component Analysis

# Results on imaginal discs

Gene & atlas



Used patterns

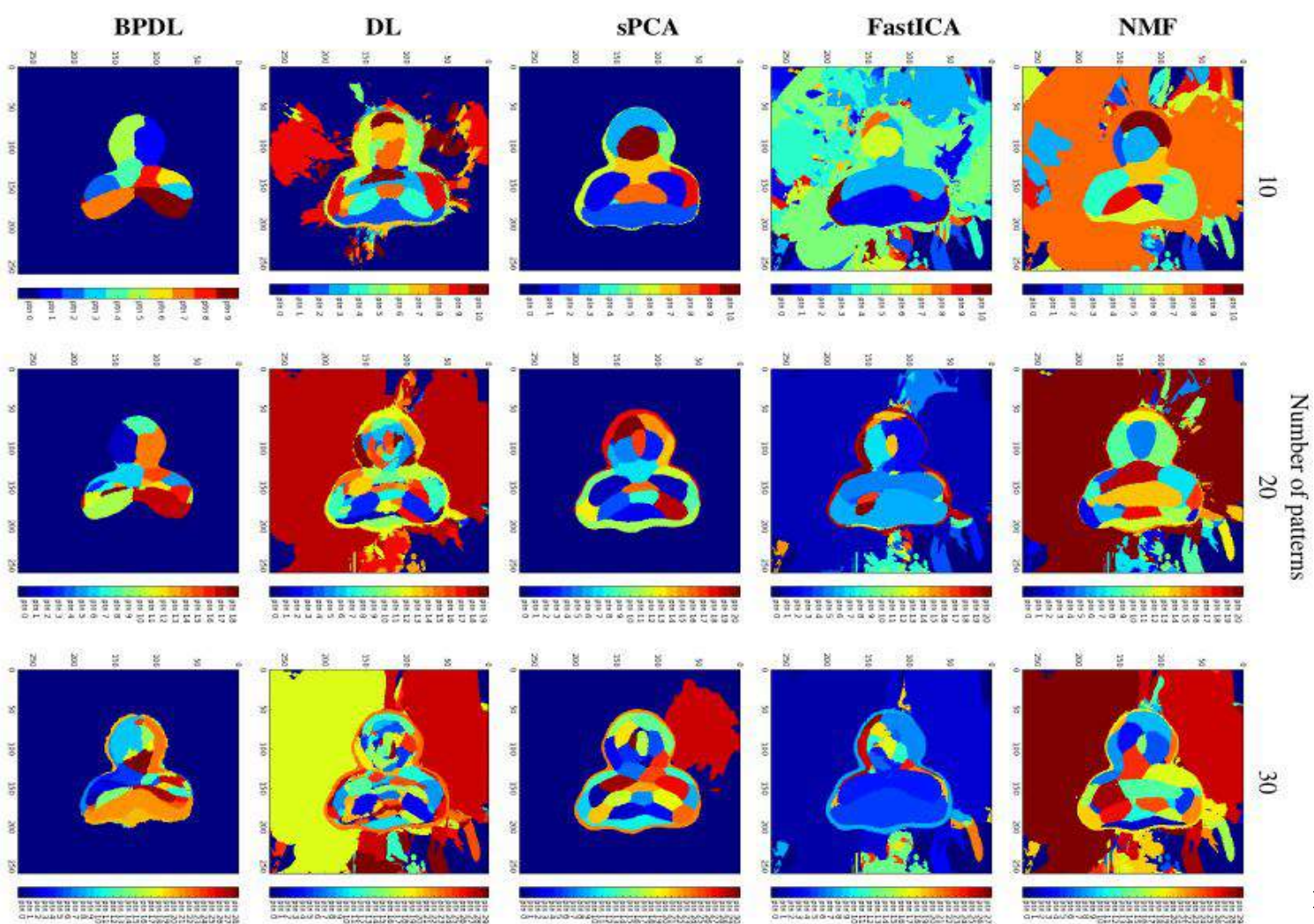


Reconstruction error  $|g - \hat{g}|$

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



# Extracted Atlases for Imaginal discs





# Conclusions

- Presented four image processing methods:
  - Image segmentation on superpixels
  - Center detection and ellipse fitting
  - Region growing with shape prior
  - Binary pattern dictionary learning
- Implementation [open-source]:
  - <http://borda.github.com/pyImSegm>
  - <http://borda.github.com/pyBPDL>
- Future work
  - Finalise complete pipeline
  - Try instance segmentation with Neural Nets
  - Complete image analysis pipeline in 2.5D
  - ...

The image shows two screenshots of GitHub project pages. The top screenshot is for 'pyImSegm', a project for 'Image segmentation - general superpixel segmentation & center detection & region growing'. It features a 'View On GitHub' button and a sidebar with a 'Image segmentation toolbox' menu. The main content area is titled 'Image segmentation toolbox' and includes a progress bar with 'BUILD' (passing), 'colorectiv' (97%), 'colorectiv' (100%), 'TUM' (shapenet), 'colorectiv' (93%), and 'PASSED'. The main text describes 'Superpixel segmentation with GraphCut regularisation' and mentions its use in medical imaging.

The bottom screenshot is for 'pyBPDL', a project for 'Binary Pattern Dictionary Learning for gene activation in microscopy images'. It also has a 'View On GitHub' button and a sidebar with a 'Binary Pattern Dictionary Learning' menu. The main content area is titled 'Binary Pattern Dictionary Learning' and includes a progress bar with 'BUILD' (passing), 'colorectiv' (93%), 'colorectiv' (100%), 'TUM' (shapenet), 'colorectiv' (93%), and 'PASSED'. The main text describes the project as a final step in an image processing pipeline for gene activation analysis. A diagram illustrates the workflow: 'Input images' are processed into 'Gene expressions' and 'Integral discs', which are then used for 'Segmentation - Registration - BPDL' to produce 'Aligned expressions into binary images' and 'Atlas'. A 'Weights' matrix is also shown.

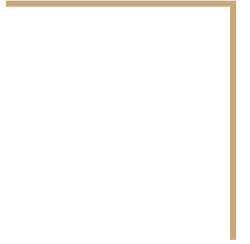
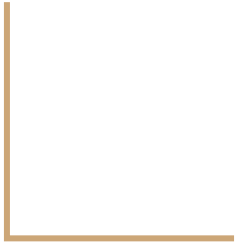
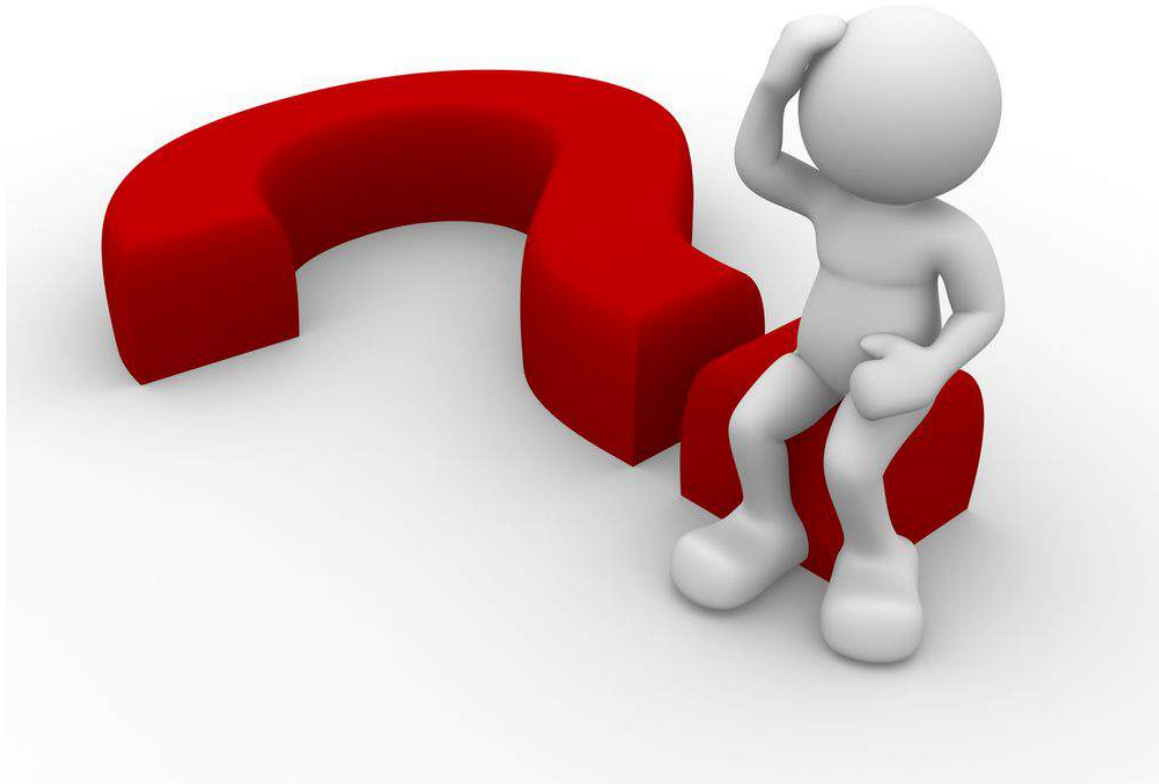
# Related Publications

## Journals:

- Borovec, J., Svihlik, J., Kybic, J., & Habart, D. (2017). **Supervised and unsupervised segmentation using superpixels, model estimation, and Graph Cut.** *Journal of Electronic Imaging*, 26(6), 26–26–17. <https://doi.org/10.1117/1.JEI.26.6.061610>
- Borovec, J., Kybic, J., & Sugimoto, A. (2017). **Region growing using superpixels with learned shape prior.** *Journal of Electronic Imaging*, 26(6), 26–26–14. <https://doi.org/10.1117/1.JEI.26.6.061611>
- Dvorak, J. et al. (2018). **Volume estimation from single images: an application to pancreatic islets.** *Image Analysis & Stereology*.
- Klema, J. et al. (2017). *Semantic biclustering for finding local, interpretable and predictive expression patterns. BMC Genomics.*

## Conferences / workshops:

- Borovec, J., & Kybic, J. (2013). **Fully automatic segmentation of stained histological cuts.** In L. Husník (Ed.), *17th International Student Conference on Electrical Engineering* (pp. 1–7). Prague: CTU in Prague.
  - The paper received a **Poster award**.
- Borovec, J., & Kybic, J. (2014). **jSLIC : superpixels in ImageJ.** In Z. Kukelova & J. Heller (Eds.), *Computer Vision Winter Workshop* (pp. 14–18).
- Borovec, J., & Kybic, J. (2016). **Binary pattern dictionary learning for gene expression representation in drosophila imaginal discs.** In *Mathematical and Computational Methods in Biomedical Imaging and Image Analysis (MCBMIIA), ACCV workshops* (pp. 555–569). Springer. [https://doi.org/10.1007/978-3-319-54427-4\\_40](https://doi.org/10.1007/978-3-319-54427-4_40)
  - The paper received a **Best paper award**.
- Borovec, J., Kybic, J., & Nava, R. (2017). **Detection and Localization of Drosophila Egg Chambers in Microscopy Images.** In Q. Wang, Y. Shi, H.-I. Suk, & K. Suzuki (Eds.), *8th International Workshop Machine Learning in Medical Imaging (MLMI)* (pp. 19–26). [https://doi.org/10.1007/978-3-319-67389-9\\_3](https://doi.org/10.1007/978-3-319-67389-9_3)



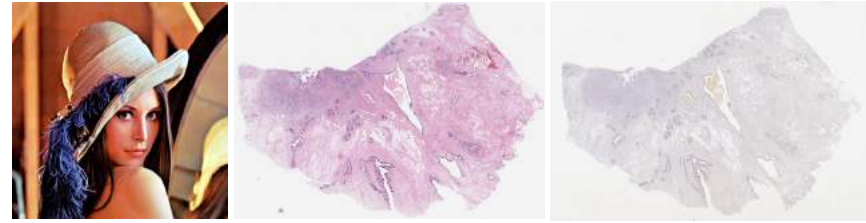
# Reviewer

**Q:** Chapter 4: Superpixel extraction:

I do not see any benefit of introducing the regularization parameter over the original one as defined in eq. (4.2).

It is claimed that the color conversion gives a 60% speed-up. Since the conversion is done only once, it is not clear to me how such a significant gain can be achieved.

Later in the thesis, different letters  $N$  and  $r$  are used for the initial superpixel size  $v$  and regularization  $\epsilon$ , respectively, which is confusing.



**Response:**

- *With the proposed regularization we expect to normalise regul. to the superpixel size which is convenient for searching segmentation hyperparameters;*
- *We use hash table and compute each color conversion only once and the colors in histopathology images are usually very repetitive;*

	unique vs total colors	unique colors vs image size
Lena	0.78%	49.63%
histo HE	0.26%	3.17%
histo Ki67	0.05%	0.59%
histo CD31	0.14%	1.93%

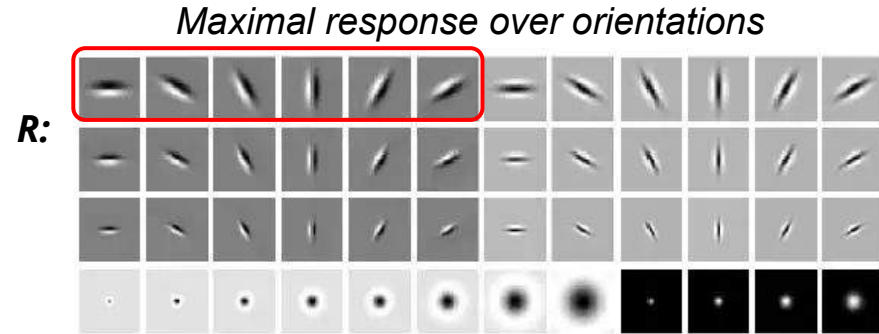
# Reviewer

**Q:** Chapter 5: (Un)supervised superpixel segmentation:

It is not clear how the number of texture features (Sec. 5.3.2) is reduced from 48 to 18.

In 5.5.1, how much different in practice is the learned binary potential  $B(k, l)$  from the simple one with zeros on diagonal are ones elsewhere?

According to Table 5.3 in Sec. 5.6.2,  $l_1$  performs equally well as  $l_\infty$ , yet  $l_\infty$  has been chosen even though this metric is generally less robust.



- *With reasonably set constant we did not observe any significant changes to learned weights;*
- *For segmenting ovaries with small appearance differences between classes we observed  $l_\infty$  as better choice, see Tab. 5.3 (evaluation in  $F_1$  score)*

Dataset	nb. classes	$l_1$	$l_2$	$l_\infty$
imaginal disc	3	<b>0.813</b>	0.808	0.807
ovary	4	0.818	0.816	<b>0.824</b>

# Reviewer

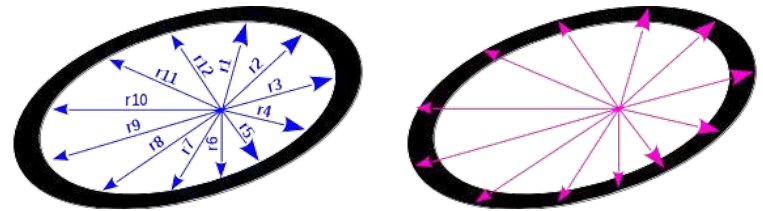
**Q:** Chapter 6: Object center detection and ellipse fitting:

Ray features (more commonly known as shape vectors) were made orientation invariant by starting with the maximum value. This is known to be an unstable approach. For example, more robust invariant can be achieved by taking the absolute value of shape vector Fourier transform.

The discussion in Sec. 6.4.1 about results in Table 6.5 is difficult to follow. What “up edge” and “down edge” means?

**Response:**

- *I appreciate the idea with absolute value of shape vector Fourier transform. It has been added to the package; We also need the inverse transform.*
- *“Up” and “Down” edge represents the interior (blue) and exterior (pink) object boundaries, respectively.*

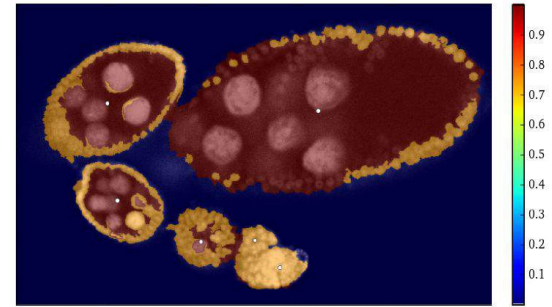


# Reviewer

**Q:** Chapter 7: Region growing with shape prior: The appearance probability map  $P_g(g_s)$  as defined in Sec. 7.1.1 would contain only values 0 or 1. Since the descriptor  $g_s \in \{1,2,3,4\}$  represents one of the four classes (cytoplasm, follicle cells, nurse cell and background), then my understanding is that  $P_g(1...3)=1$  and  $P_g(0)=1$ .

In Fig. 7.2, are (a) and (c) related by a simple transformation from polar to Cartesian coordinates?

**R:**



- *The  $P_g(g_s)$  is a probability of really being a foreground / background given the estimated semantic segmentation;*

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

- *Yes, for cumulative distribution.*

# Reviewer

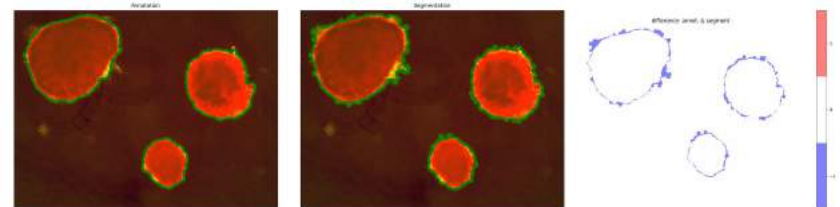
**Q:** Chapter 7: Region growing with shape prior: In Sec. 7.2 it is mentioned that the binary Graph Cut algorithm is preferred since a global minimum is guaranteed in this case. However, it is important to note that the whole region growing algorithm (Algorithm 1) alternates between two steps: shape parameter update and label update. It is therefore a type of EM algorithm with local minima and it is not guaranteed that the global minimum is reached anyway. Reaching the global minimum in the label update step using the binary GC may not improve anything.

This chapter is in my opinion the main contribution of the thesis and it would be beneficial to explain and even demonstrate how this approach can be used as a general segmentation technique.

## **Response:**

- *Comparison with previous (un)supervised segmentations on Langerhans islets, RG uses GMM foreground modeling with user center initializations.*

[67 images]	mean ARS	mean $F_1$	min $F_1$
<b>GMM &amp; GC</b>	0.860±0.061	0.939±0.031	0.820
<b>GMM &amp; RG</b>	0.890±0.033	0.953±0.017	0.905





# Reviewer

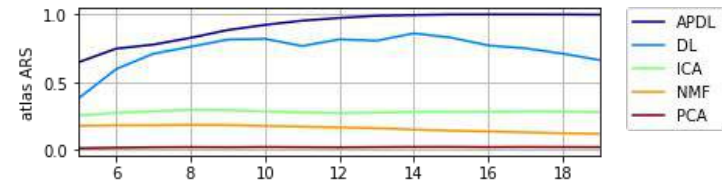
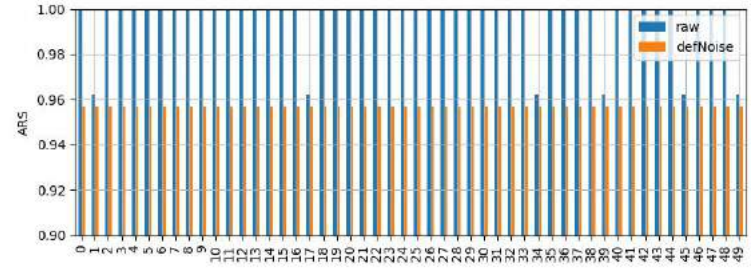
**Q:** Chapter 8: Binary pattern dictionary learning: Dictionary learning clearly depends on the number of partners  $K$ , but it can also depend on the dictionary initialization. Was this possibility considered?

Results in Fig. 8.4 illustrate that for a given  $K$  different methods provide totally different patterns. This could be an indication that the optimization is not stable in practice. Is there any follow-up research that would prove that this method is useful in biology for observing gene expressions?

[1] Klema, J. et al. (2017). Semantic biclustering for finding local, interpretable and predictive expression patterns. BMC Genomics.

**R:**

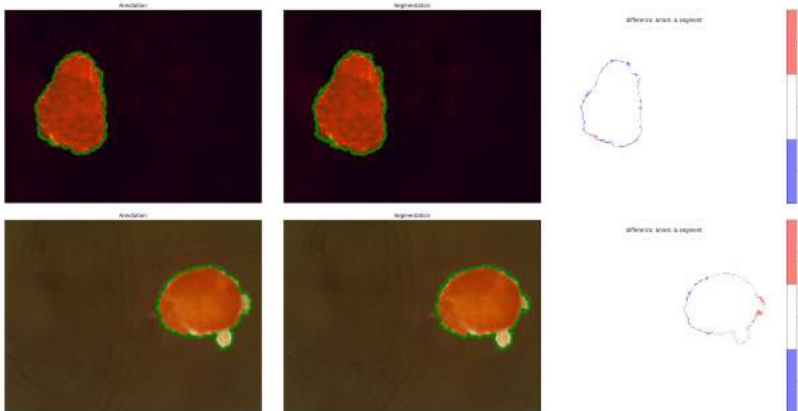
- *Randomness has been addressed by repeated random initialisations;*
- *Results on imaginal discs are in Chapter 8.4.3 with application in [1];*
- *Results on Drosophila ovaries are a work in progress;*
- *For synthetic data BPDFL shows robustness to various number of patterns.*



# Reviewer

## Remark:

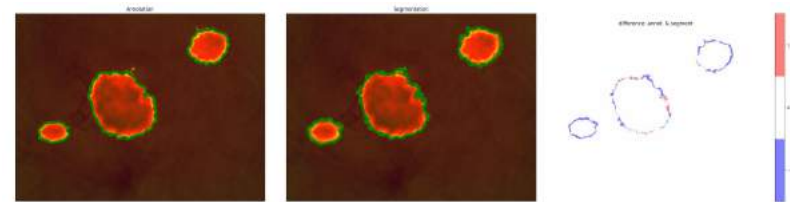
It would be interesting to see the type of scientific questions and conclusions that could be inferred using Mr. Borovec's tools.



## Response:

*Recently we used segmentation as preprocessing for estimation of Langerhans islets [1].*

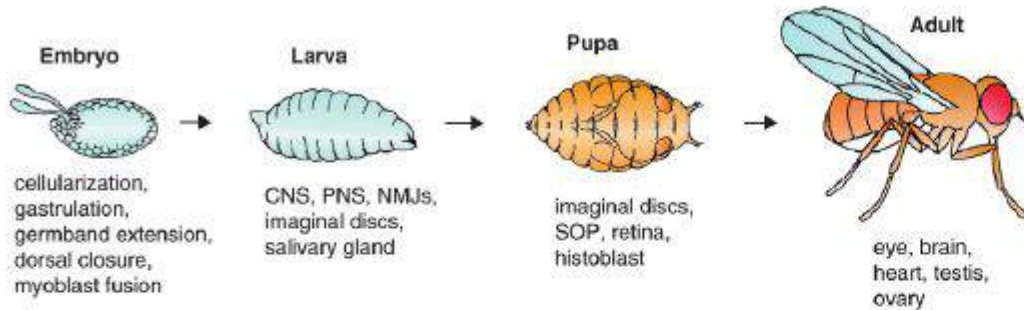
[67 images]	ARS	mean $F_1$	min $F_1$
<b>RF &amp; GC</b>	0.938±0.050	0.974±0.018	0.859



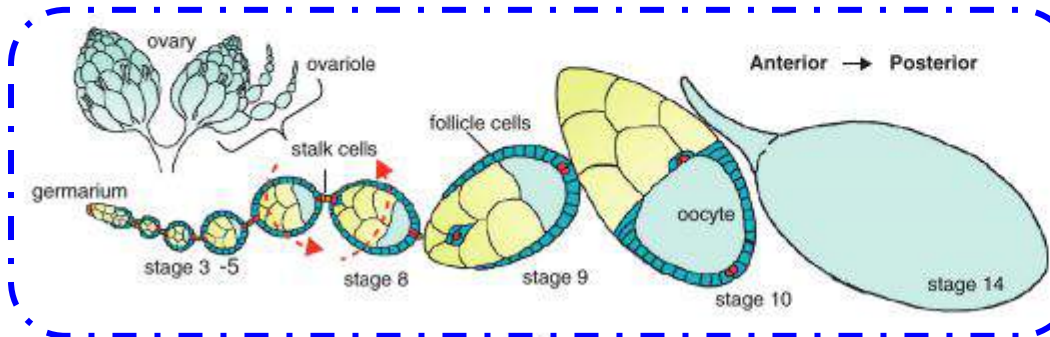
[1] Dvorak, J. et al. (2018). **Volume estimation from single images: an application to pancreatic islets.** Image Analysis & Stereology.

# Appendix

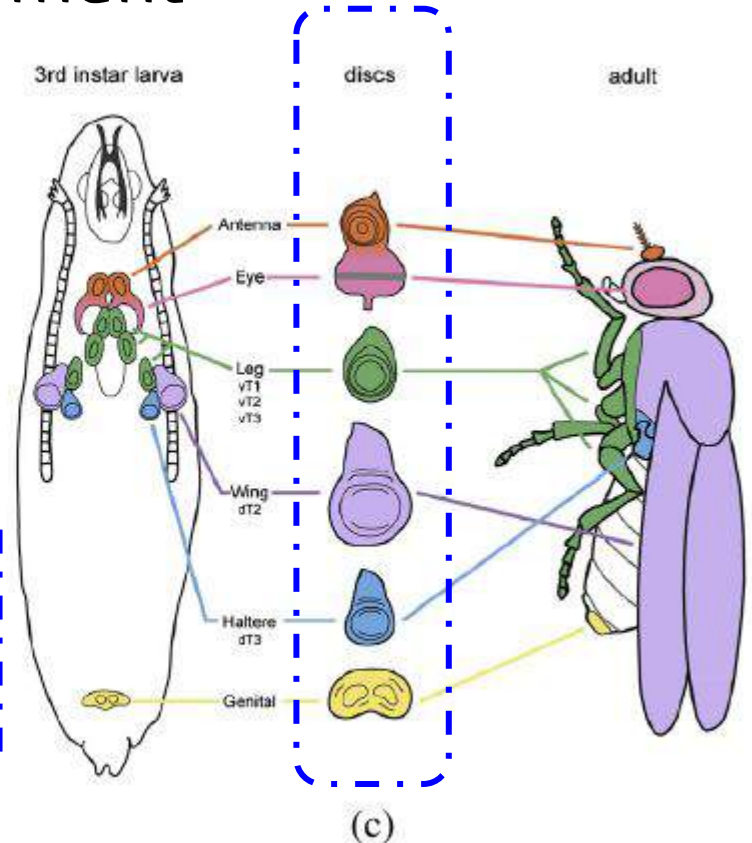
# Schema of Drosophila development



(a)

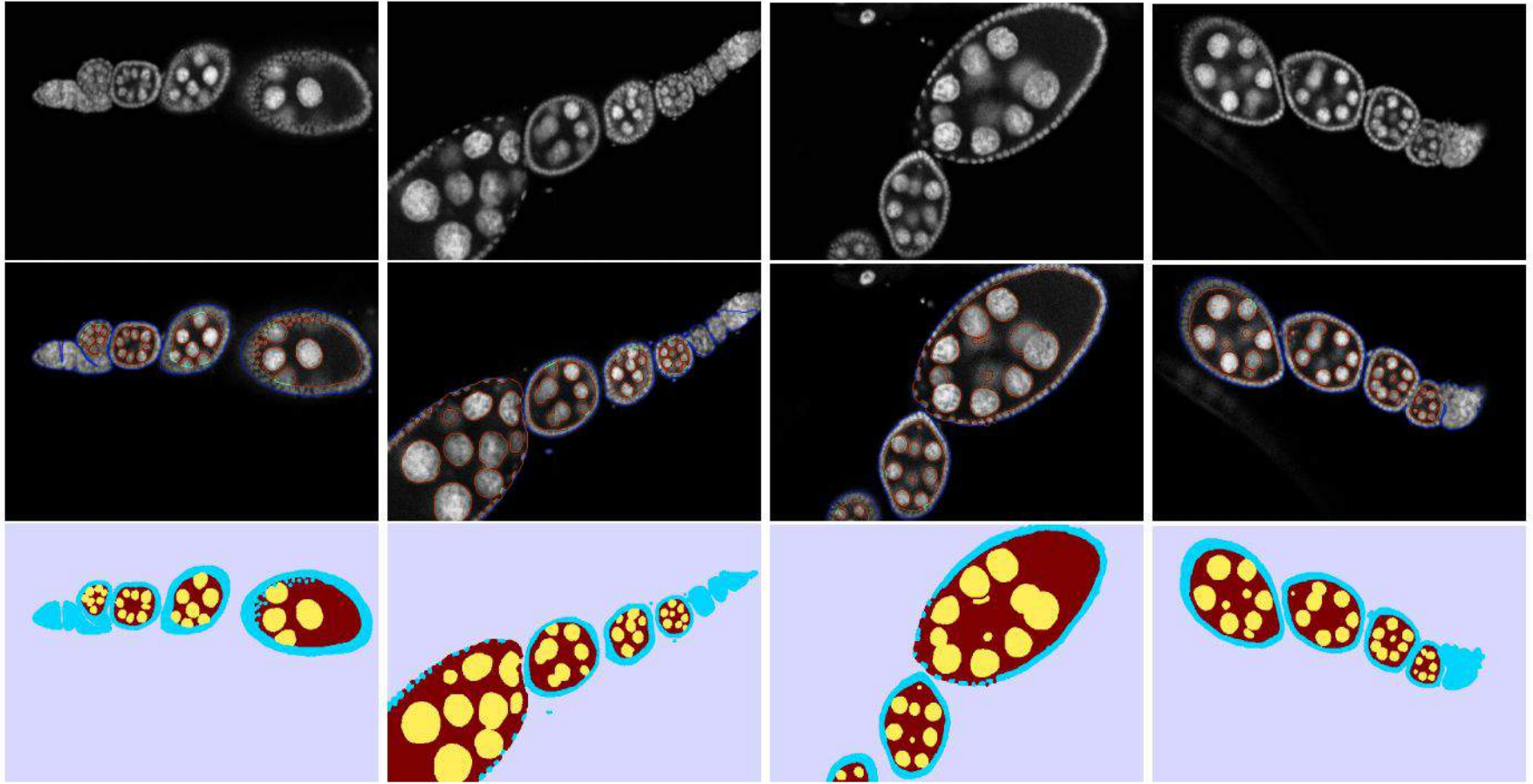


(b)

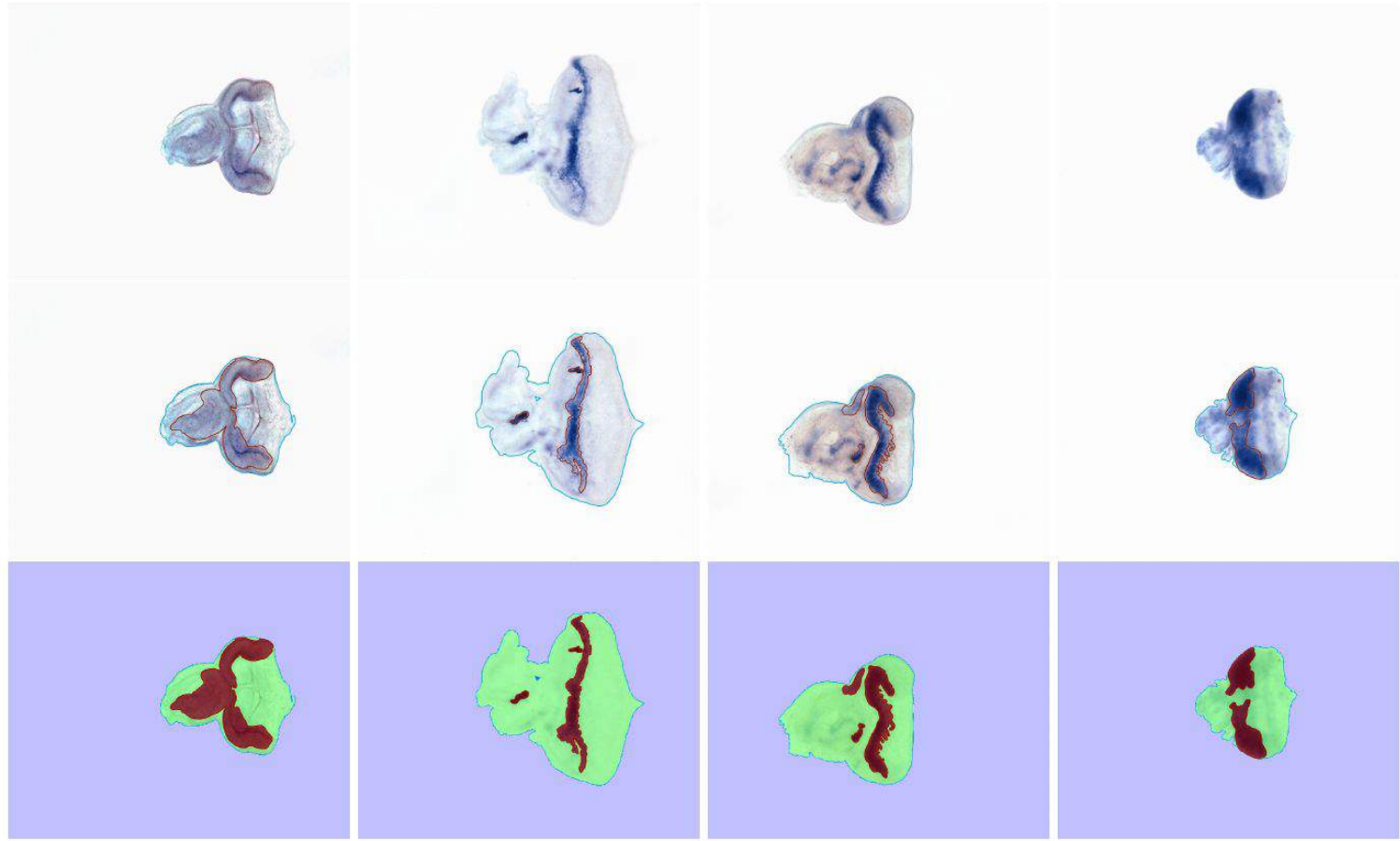


(c)

# Drosophila ovary - semantic segmentation



# Drosophila imaginal discs - semantic segmentation



# Notations

## Image related

$\Omega$	set of pixels (image plane)
$I$	input image function $I : \Omega \rightarrow \mathbb{R}^m$
$L$	set of labels

## Superpixels

$\eta$	original regularization
$\xi$	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
$\Omega_s$	pixels belonging to a superpixel $s, \Omega_s \subset \Omega$

## Segmentation & Region growing

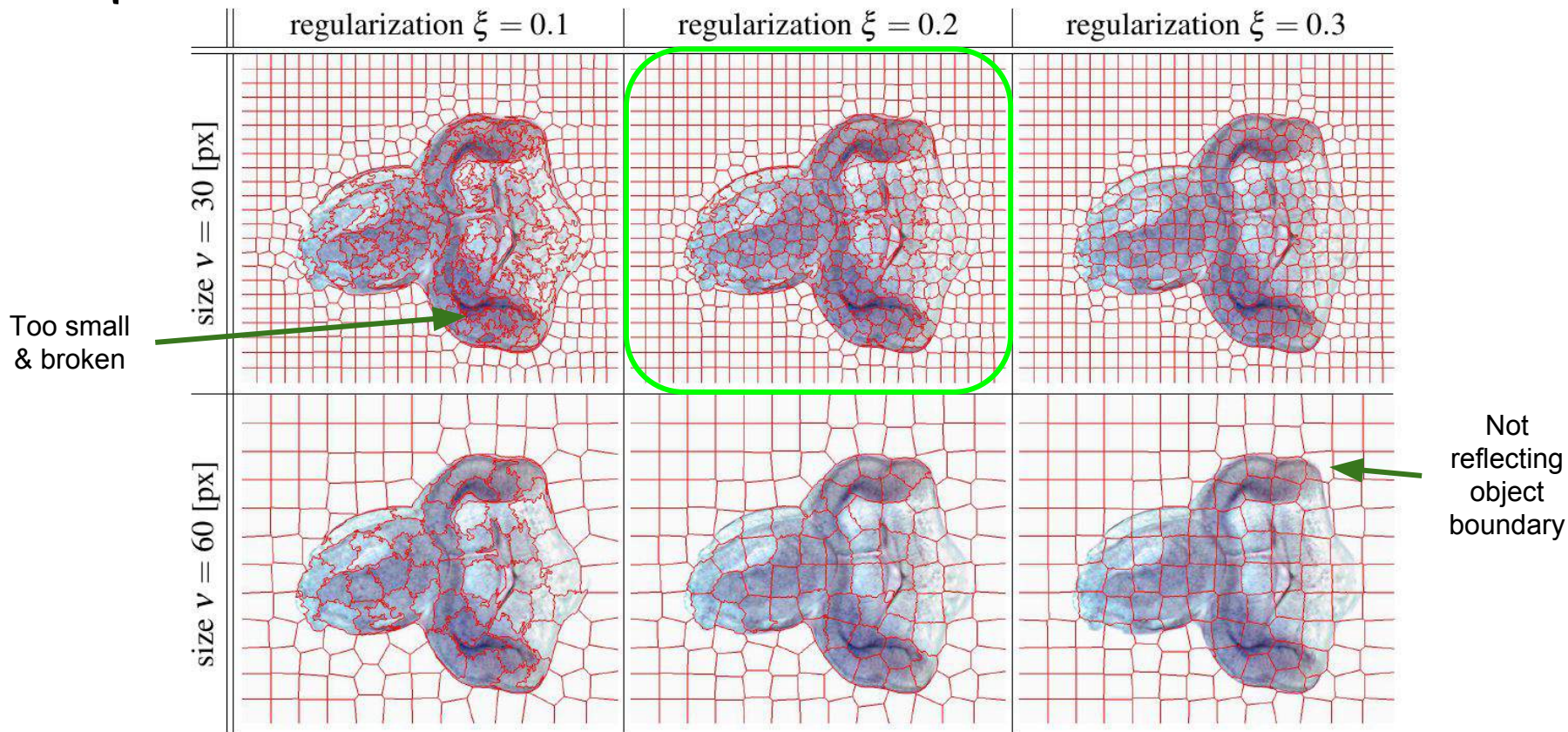
$y_\Omega$	pixel-wise segmentation function $y_\Omega : \Omega \rightarrow L$
$Y_\Omega$	ordered set of pixel-wise segmentation $Y_\Omega = y_\Omega(\Omega)$
$y$	superpixel segmentation function $y : S \rightarrow L$ with abbrev. for $y_s = y(s)$
$Y$	ordered set of superpixel segmentation $Y = y(S)$
$x$	feature vector
$X$	set of features $x_s \in X$ 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 $g \in G$
$y_\Omega, Y_\Omega$	atlas (binary patterns, segmentation)
$w$	vector with binary weights
$W$	matrix with binary weights $w \in W$

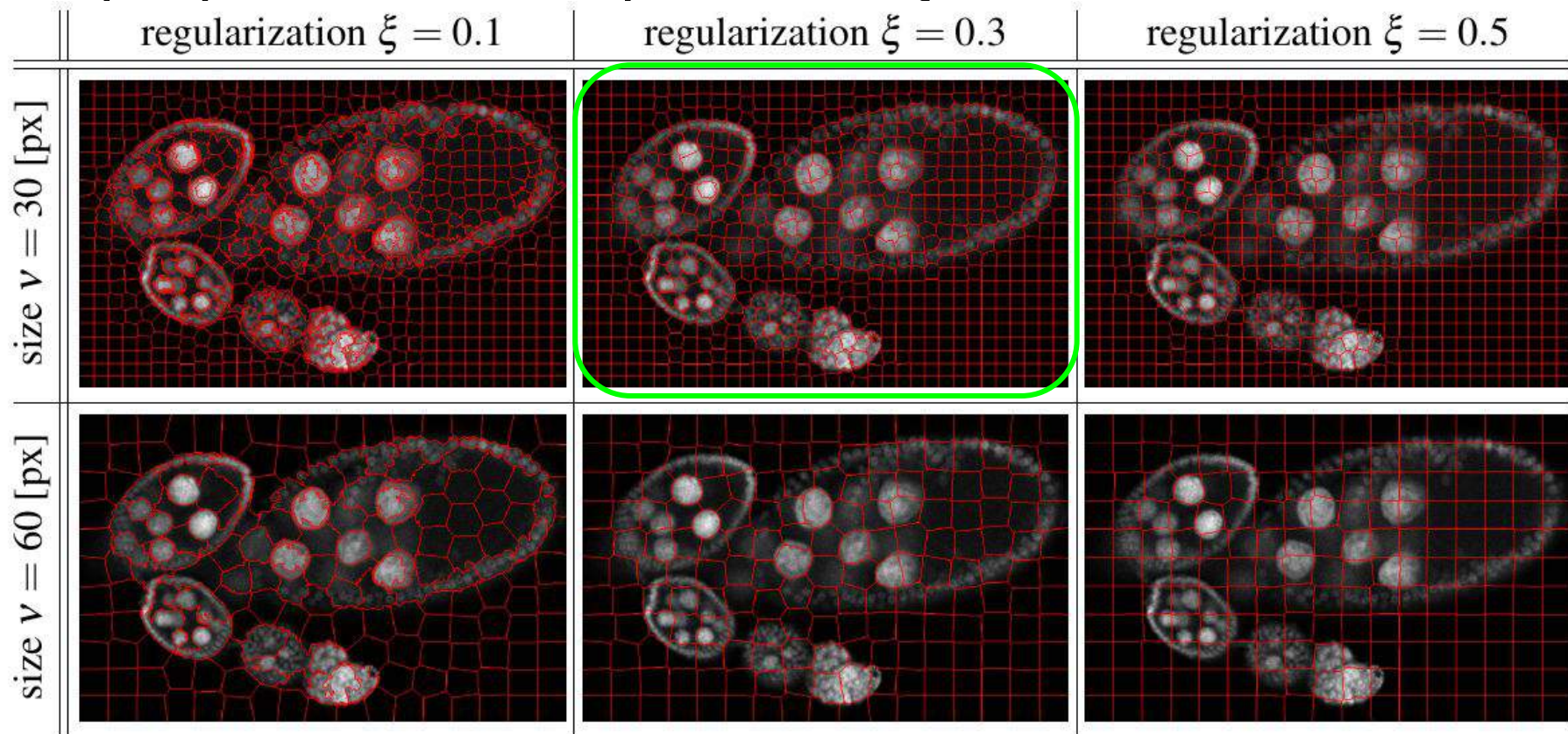


# Superpixels - Imaginal discs





# Superpixels - Drosophila ovary



# Problem formulation

Formulation as maximum a posteriori for segmentation  $Y$  and features  $X$  with pairwise regularisation  $R$

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

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

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

Energy minimisation

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

# 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

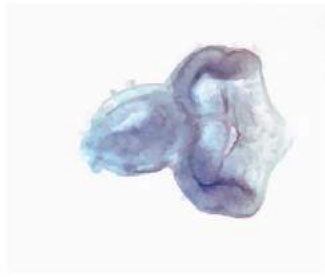
# Superspixel features

Superspixel color means

input image



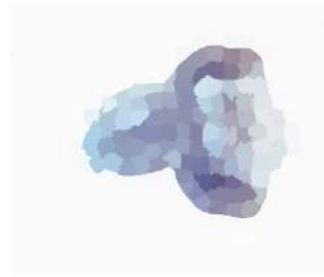
$N = 15$  pixels



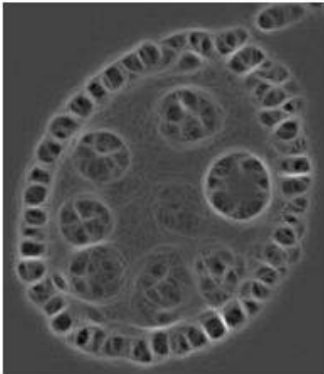
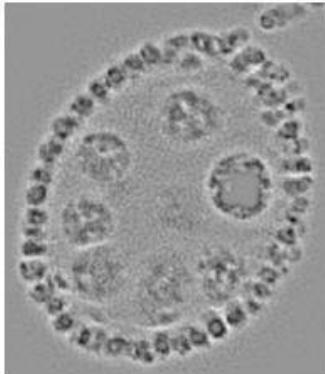
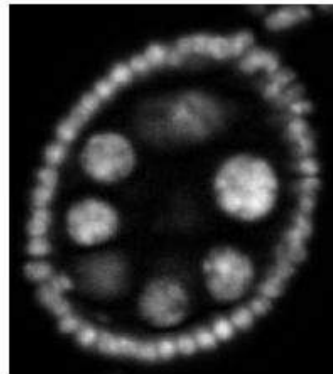
$N = 30$  pixels



$N = 45$  pixels



Leven-Malik filter bank





# Graph Cut - Edge weights

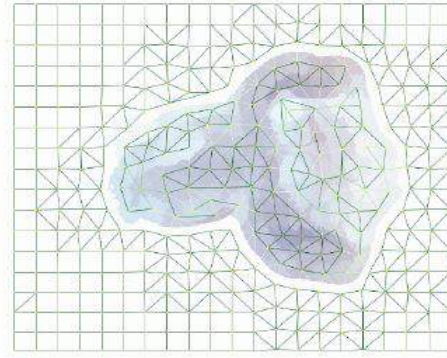
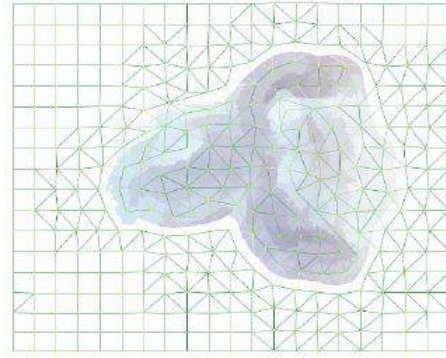
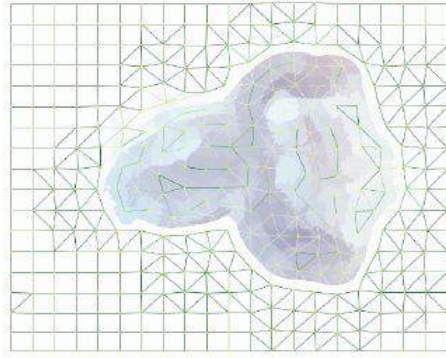
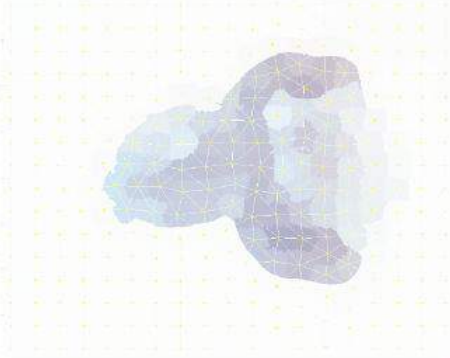
Spatial

Color

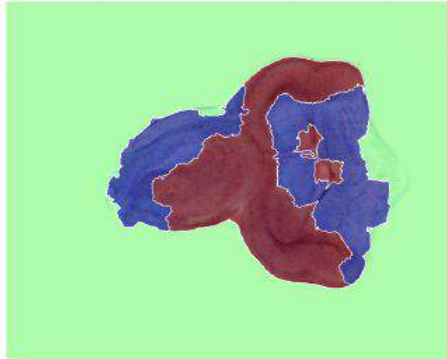
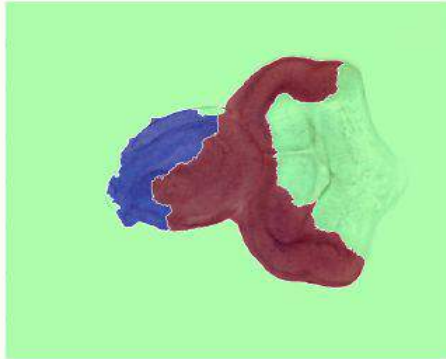
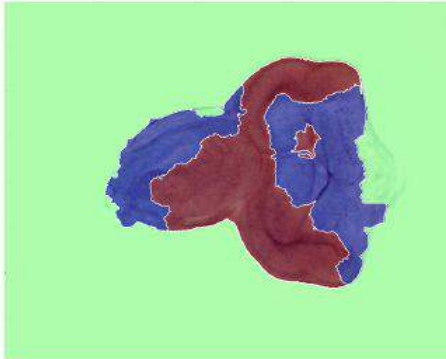
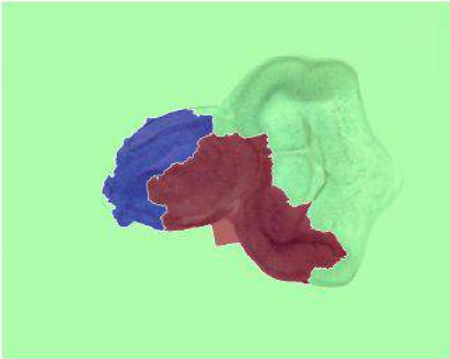
Features

Models

weighted edges



segmentation

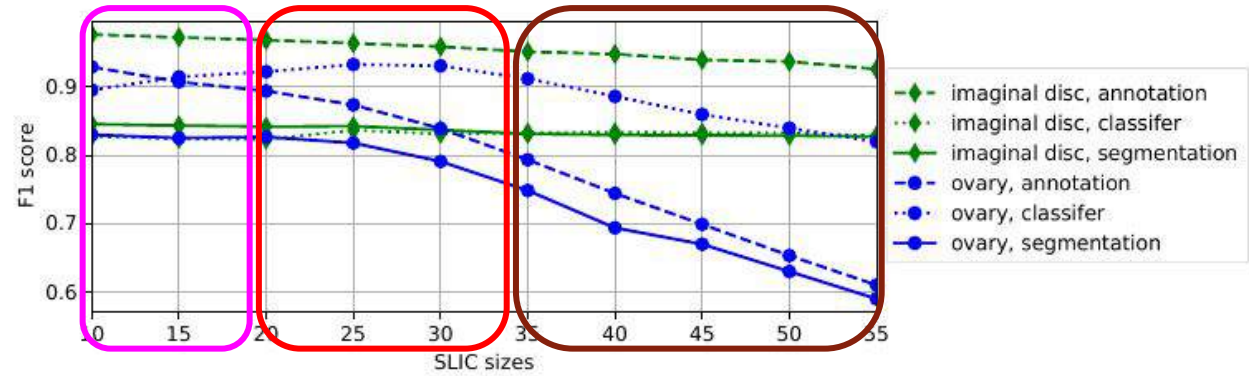


# Influence of superpixel parameters

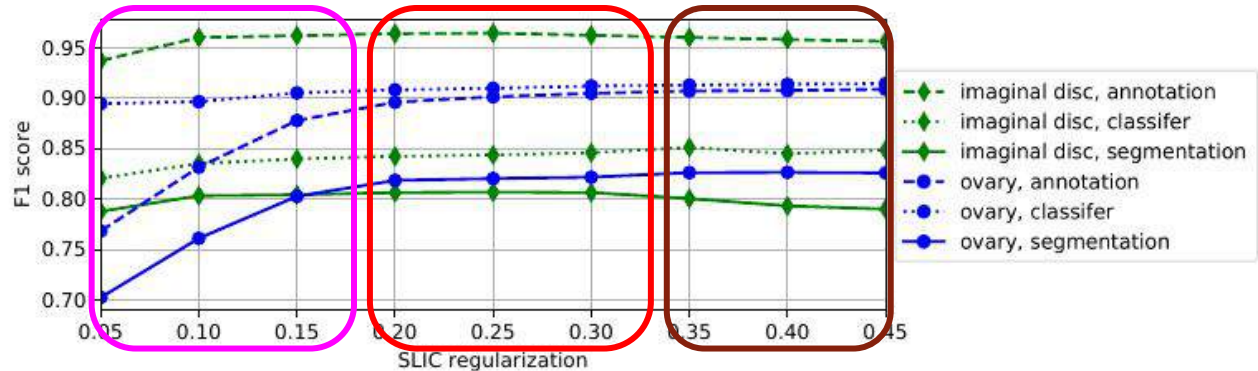
Low feature  
representativeness

Optimum  
(~compromise)

Low separability



(a)





# Ellipse fitting

Formulate as likelihood maximization

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

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

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

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

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

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

# Region growing - variational framework

Where:

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

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

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

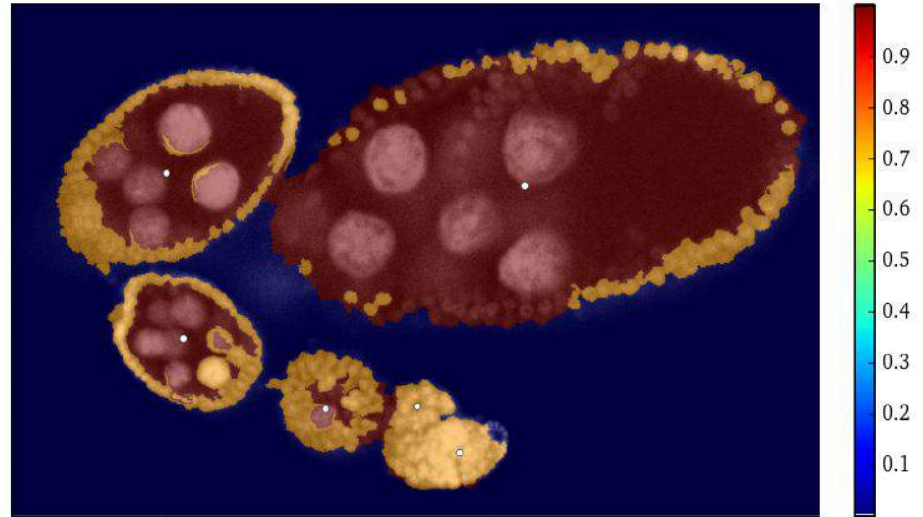
Leads to energy minimisation:

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

# Appearance model

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

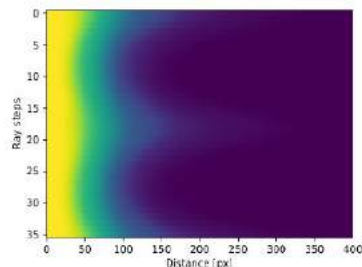
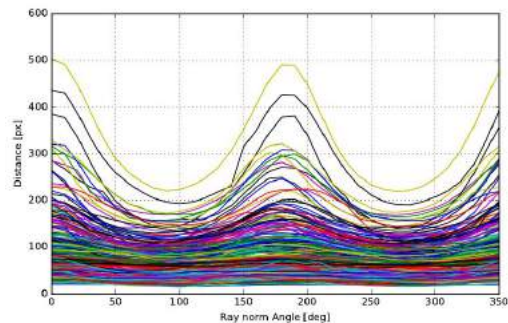
$$P_g(\mathbf{y}(s)|g_s) = \begin{cases} P_g(g_s) & \text{for } \mathbf{y}(s) \neq 0 \\ 1 - P_g(g_s) & \text{for } \mathbf{y}(s) = 0 \end{cases}$$



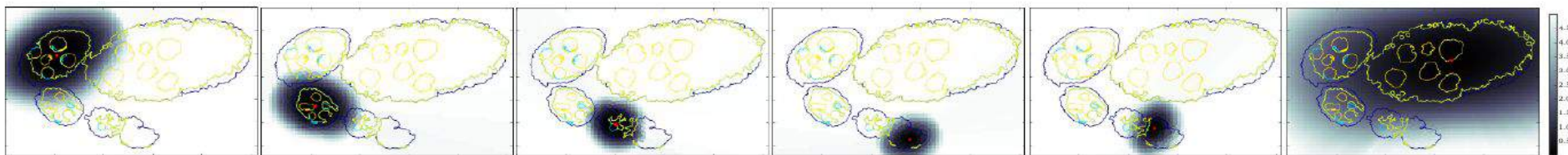
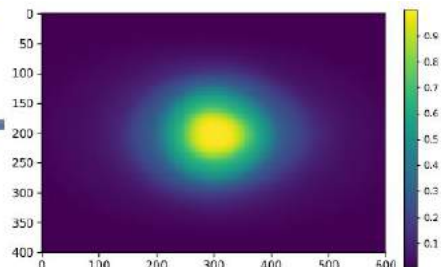
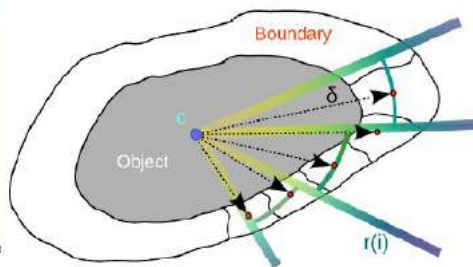
# Shape model & prior

Joint gaussian model of radial (ray) distances

Probability of boundary being at a point or further

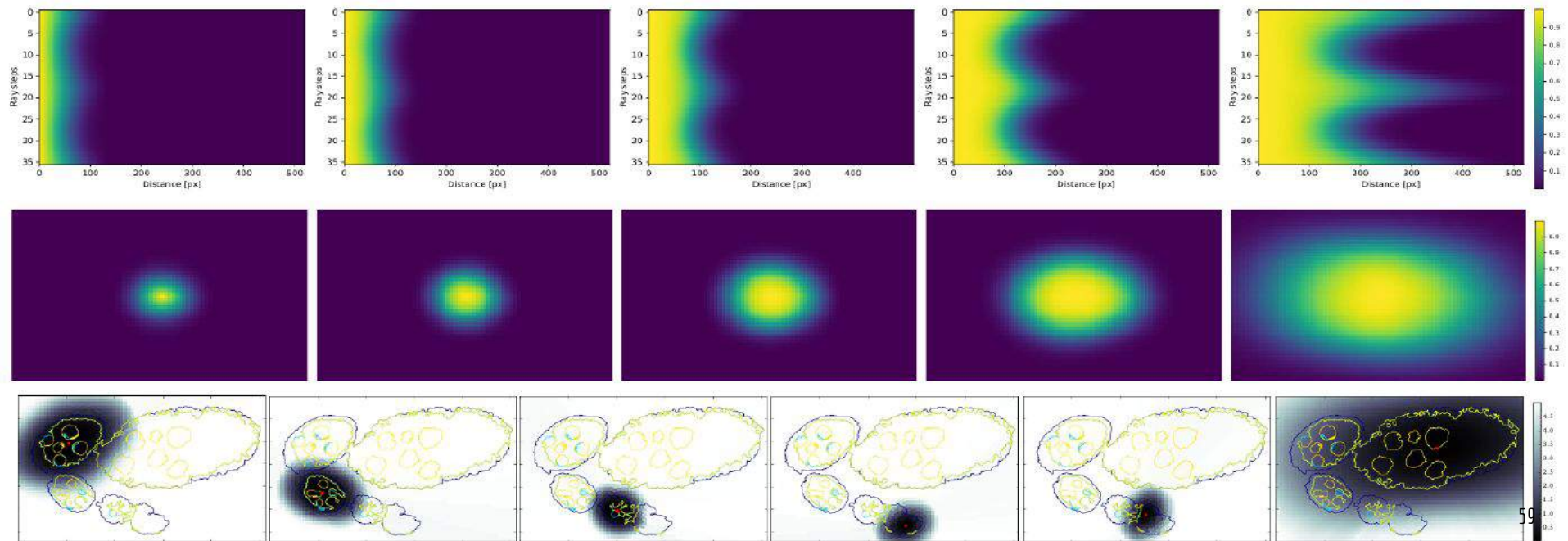


$$q(s, m_k) = \int_{\delta}^{\infty} \rho(r) dr = 1 - \int_0^{\delta} \rho(r) dr$$



# Mixture of Gaussian models

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



---

**Algorithm 1:** Region growing.

---

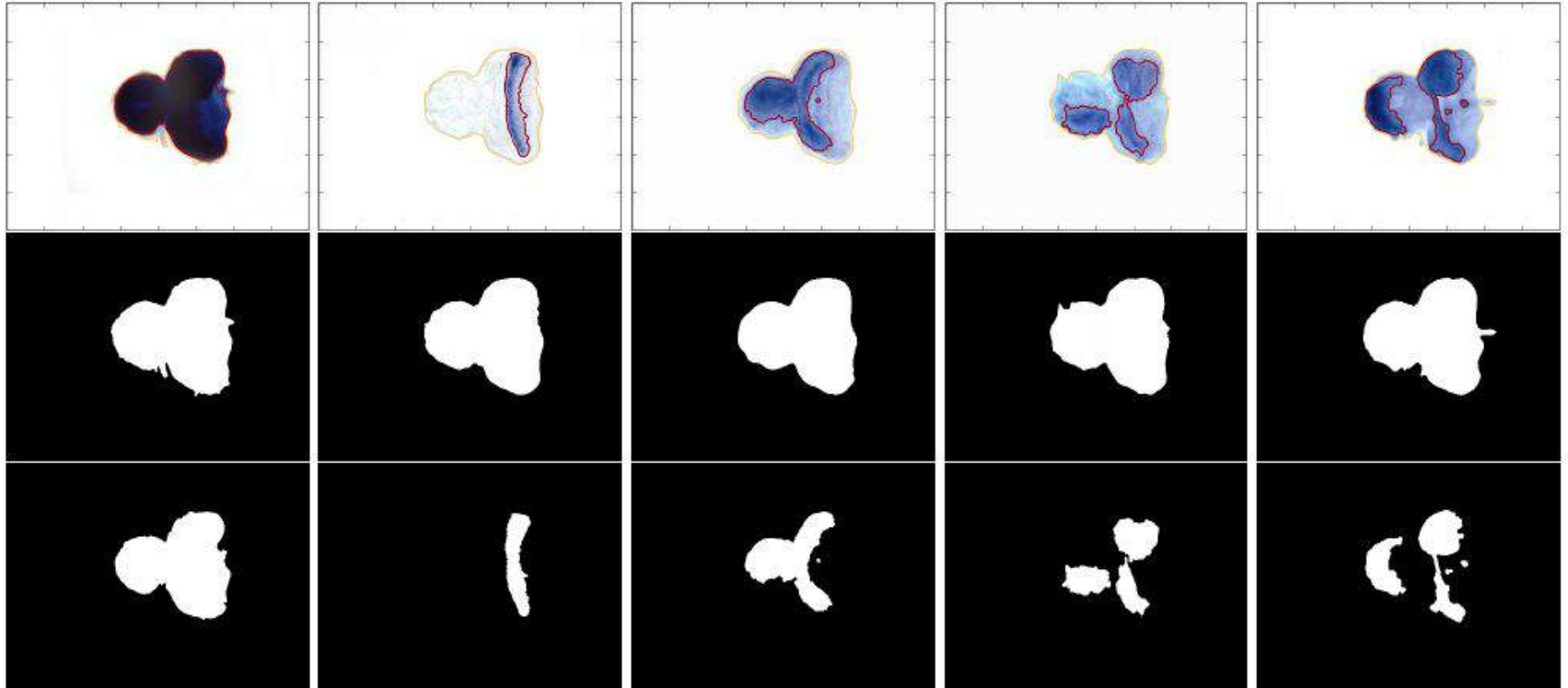
**Input:**  $S$ : superpixels,  $g$ : superpixel descriptors,  $c_k$ : initial object centers,  $M$ : mixture of statistical shape models

**Output:** object segmentation  $\mathbf{y}$

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



# Input segmented imaginal discs



# Algorithm

---

**Algorithm 1** General schema of BPDFL 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**
- 

*It makes the algorithm more robust to initialisation.*

# Comparison on synth. images

datasets		NMF	FastICA	sPCA	DL	BPDL
<b>v2</b>		<i>(size 128 × 128 px, 23 patterns)</i>				
<i>pure</i>	ARS	<b>1.0</b>	<b>1.0</b>	<b>0.989</b>	<b>1.0</b>	<b>0.999</b>
	diff.	<b>0.0</b>	<b>0.0</b>	0.037	<b>0.0</b>	<b>0.005</b>
	time [s]	<b>82.329</b>	5533.4	460.82	14786.	<b>88.260</b>
<i>deform</i>	ARS	0.818	<b>0.846</b>	0.801	0.807	<b>0.970</b>
	diff.	0.019	<b>0.015</b>	0.056	0.046	<b>0.004</b>
	time [s]	<b>144.10</b>	5683.2	477.47	13619.	<b>165.22</b>
<i>D&amp;N</i>	ARS	0.120	<b>0.612</b>	0.024	0.144	<b>0.877</b>
	diff.	<b>0.036</b>	<b>0.036</b>	0.092	0.039	<b>0.013</b>
	time [s]	<b>77.399</b>	6912.9	485.44	13729.	<b>289.51</b>

# Parameter selection

# & More discs

