Active Graph Matching for Automatic Joint Segmentation and Annotation of *C. elegans*

Dagmar Kainmueller[1], Florian Jug[1], Carsten Rother[2], and Gene Myers[1]

[1] Max Planck Institute of Molecular Cell Biology and Genetics, Germany
[2] Computer Vision Lab Dresden, Technical University Dresden, Germany
kainmueller@mpi-cbg.de

Abstract. In this work we present a novel technique we term *active graph matching*, which integrates the popular active shape model into a sparse graph matching problem. This way we are able to combine the benefits of a global, statistical deformation model with the benefits of a local deformation model in form of a second-order random field. We present a new iterative energy minimization technique which achieves empirically good results. This enables us to exceed state-of-the art results for the task of annotating nuclei in 3D microscopic images of *C. elegans*. Furthermore with the help of the generalized Hough transform we are able to jointly segment and annotate a large set of nuclei in a fully automatic fashion for the first time.

1 Introduction

A frequently used model organism in developmental biology is the worm *C. elegans*. Since *C. elegans* is highly stereotypical it is well suited for comparative developmental studies. A common and time consuming problem in such studies is the segmentation and annotation of cell nuclei with their unique biological names in 3D microscopic images [1,2,3,4]. This work presents a fully automated joint segmentation and annotation method for this task.[1]

Previous approaches for automatic annotation of nuclei in *C. elegans* [2,3,4] build an average atlas of nuclei locations, and annotate new target worms by mapping the atlas to the target. One approach finds a globally optimal one-to-one mapping but is agnostic to covariances between nucleus positions [2,3]. Another approach incorporates heuristic prior knowledge on relative positions of nuclei in the form of a local deformation model, but does not tackle local deformation and mapping of nuclei jointly in a globally optimal way [4].

While these approaches are in principle capable of performing segmentation and annotation of nuclei in a fully automatic manner in images that show all nuclei of *C. elegans* L1 larvae, none of them has been evaluated quantitatively in this respect: Only the annotation of given, correct segmentations is evaluated quantitatively in [2], and only images which exclusively show the 80 widely spaced *body wall muscles* of *C. elegans* L1 larvae are considered in [4].

---

[1] We work exclusively with *disentangled, straightened* images of *all 558 nuclei of L1 larvae*. Disentangling and straightening are not topics of this paper. See e.g. [5,6].
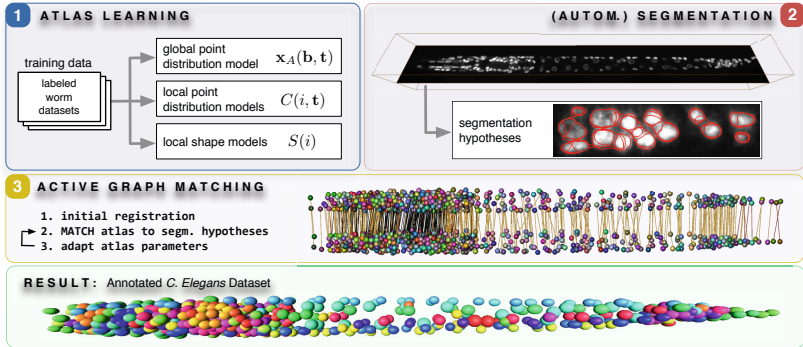
Fig. 1. A sketch of the proposed pipeline. First, a statistical atlas is learned from annotated data. New, straightened images are then segmented automatically. Subsequently, the body axes of atlas and segmentation are aligned. *Active Shape Matching* then alternates between graph matching and optimization of atlas parameters.

We propose to *learn* from training data a global-plus-local deformation model of *C. elegans*. The global model serves as a "backbone" of plausible nuclei constellations, while the local model penalizes deviations from this backbone. We use an *active shape model* [7] as a global model, and exploit the 2nd moments of the learned distribution of relative locations of nuclei for a local model.

We achieve a one-to-one mapping between atlas and target nuclei which is *approximately globally optimal* w.r.t. the local deformation model with an advanced *graph matching* method [8]. However, our overall goal is to find a one-to-one mapping which is optimal w.r.t. the local *as well as the global* deformation model. Our main technical contribution is a strong optimization technique for this problem. It can be seen as a generalization of the Iterative-Closest-Points method, where "Closest" (referring to local, point-wise measures) is replaced by "Best Matching" in terms of a global 2nd order matching energy. To the best of our knowledge only one work [9] has presented a related idea, but active shape models are not considered.

To summarize, our main contributions are (i) a new model we term *active graph matching* with an associated optimization technique, and (ii) an experimental validation that such a complex model can be optimized successfully for fully automatic segmentation and annotation of nuclei in *C. elegans* L1 larvae. Our method is, to our knowledge, the first fully automatic method ever evaluated quantitatively for this problem. Furthermore it considerably outperforms the state-of-the-art for annotation of given manual segmentations [2]. Finally, a small contribution is the idea of *in-painting* missing nuclei into the training set.

2   Method and Data

Figure 1 sketches the proposed pipeline. It builds upon a *statistical atlas* of *C. elegans* learned from training data. (cf. Sec. 2.1). Given a new target image,
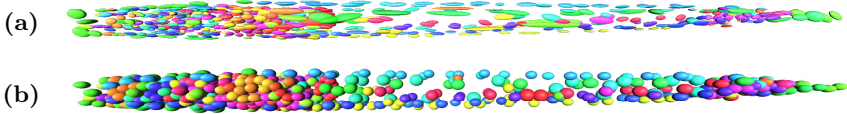
Fig. 2. Two types of local statistical models in our *C. elegans* atlas. **(a)** Local point distribution models per nucleus: Respective covariance matrices $C(i, I)$ represented as ellipses. **(b)** Average shape $s_A(i)$ of each nucleus in the atlas. Note that the local shape models also contain the respective covariances. Figures show the *in-painted* atlas.

first, a set of segmentation hypotheses is generated (cf. Sec 2.2). Then the *body axes* of the worm are determined for rough initial alignment of the atlas (cf. Section 2.3). An *objective function* encodes the problem of matching the atlas to the segmentation hypotheses. We optimize the objective by *Active Graph Matching* (cf. Sec. 2.3), a novel "iterative best matching points" method that alternates between optimal graph matching and optimal adaptation of global atlas parameters. The matching selects a subset of segmentation hypotheses, while simultaneously annotating them with their biological names.

We use a set of *training worms* also used in [2].[2] All nuclei that can be distinguished by eye were segmented manually in 30 images of *C. elegans* L1 larvae. Per worm 357 nuclei were annotated manually. See [2] for more details on the data. Let n_W denote the number of training worms, and n_A the number of annotated nuclei. Per worm, index i, with $1 \leq i \leq n_A$, represents the i-th annotated nucleus. Nuclei are sorted consistently by their biological names.

2.1   Atlas

Our statistical atlas of *C. elegans* consists of (i) a global model of the variability of the *point clouds* formed by each worm's nuclei center points, (ii) local models of the variability of single nucleus locations (cf. Figure 2a), (iii) local models of the variability of the shape of each nucleus (cf. Figure 2b), and (iv) local models of the variability of offsets between any two nucleus locations.

Global Point Distribution Model. From the set of training worms we extract locations of nuclei center points. We denote the center point location of nucleus $i \leq n_A$ in training worm $w \leq n_W$ as $x(i, w) \in \mathbf{R}^3$, the concatenation of all locations of worm w as $\mathbf{x}_w := (\ldots, x(i, w)^T, \ldots)^T \in \mathbf{R}^{3n_A}$, and the matrix of all training vectors as $\mathcal{X} := (\ldots, \mathbf{x}_w, \ldots)$. From the set of training vectors $\mathbf{x}_w$, we build a point distribution model [7] of nuclei locations. Therefore we align all training vectors via Procrustes analysis, and then perform principle component analysis, yielding the eigenvectors $\mathbf{p}_k \in \mathbf{R}^{3n_A}$ of the covariance matrix $(1/(n_W - 1))(\mathcal{X} - \bar{\mathcal{X}})(\mathcal{X} - \bar{\mathcal{X}})^T$, where $\bar{\mathcal{X}}$ denotes a matrix with the average nuclei location vector $\bar{\mathbf{x}}_A := (1/n_W) \sum_w \mathbf{x}_w$ in every column. We denote the matrix assembled from the eigenvectors as $P := (\ldots, \mathbf{p}_w, \ldots)$. The point distribution model can then be formulated as $\mathbf{x}_A(\mathbf{b}, \mathbf{t}) := \mathbf{t}(\bar{\mathbf{x}}_A + P \cdot \mathbf{b})$ with $\mathbf{b}$ denoting a vector of global

---

[2] We thank H. Peng, F. Long, X. Liu and S. Kim for providing the data.

shape parameters, and $\mathbf{t} : \mathbf{R}^3 \to \mathbf{R}^3$ an affine transformation. $\mathbf{t}$ consists of a 3x3 matrix R, and an offset vector $\mathbf{o}$. Per nucleus i, the model reads $x_A(i, \mathbf{b}, \mathbf{t}) := R(\bar{x}_A(i) + P(i) \cdot \mathbf{b}) + \mathbf{o}$, with $\bar{x}_A(i)$ denoting the average location of nucleus i, and $P(i)$ denoting rows $3i - 2$, $3i - 1$ and $3i$ of P. Each training vector can be represented by some global shape parameter vector: $\exists \mathbf{b}_w : \bar{\mathbf{x}}_A + P \cdot \mathbf{b}_w = \mathbf{x}_w$. In our global shape model we confine the shape parameters to the min/max respective values that appear in training data.

Local Models. As for our local point distribution model: Let $\mathcal{X}(i)$ denote the matrix with all training locations of nucleus i as columns. The covariance matrix per nucleus location, $C(i, R) := (1/n_W)R(\mathcal{X}(i) - \bar{\mathcal{X}}(i))(\mathcal{X}(i) - \bar{\mathcal{X}}(i))^T R^T \in \mathbf{R}^{3 \times 3}$, allows us to measure the distance of some point $x \in \mathbf{R}$ to location i in the atlas: $\text{locDiff}(x, i, \mathbf{b}, \mathbf{t}) := (x - x_A(i, \mathbf{b}, \mathbf{t}))^T \cdot C(i, R)^{-1} \cdot (x - x_A(i, \mathbf{b}, \mathbf{t}))$.

We describe the shape of an individual nucleus by means of the radii of an ellipsoid fit to the nucleus volume, sorted by value. We denote the shape of nucleus i in training worm w as $s(i, w) \in \mathbf{R}^3$. From the training data, we derive the average shape per nucleus, $s_A(i)$, as well as the respective covariance matrix $S(i)$. Thus we can measure the distance of some shape s to the shape of atlas nucleus i as $\text{shapeDiff}(s, i) := (s - s_A(i))^T \cdot S(i)^{-1} \cdot (s - s_A(i))$.

In addition to nucleus-individual statistics, we also perform statistics on offset vectors between any two nuclei: Let $d(i, j, w) := x(i, w) - x(j, w)$ denote a training offset vector. We retrieve the average offset $\bar{d}_A(i, j) := (1/n_W) \sum_w d(i, j, w)$ as well as the respective covariance matrix $D(i, j, R)$. Let $d_A(i, j, \mathbf{b}, R)$ denote an offset vector in an instance of the global point distribution model. Then, we can measure the distance of some offset d w.r.t. nuclei i and j in the atlas: $\text{offsetDiff}(d, i, j, \mathbf{b}, R) := (d - d_A(i, j, \mathbf{b}, R))^T \cdot D(i, j, R)^{-1} \cdot (d - d_A(i, j, \mathbf{b}, R))$.

Furthermore the determinant of the covariance of offsets, $\det D(i, j, R)$, lets us measure how closely two nuclei locations correlate, and thus lets us define a K-neighborhood on the atlas, denoted as $\mathcal{N}_K$. This neighborhood contains all pairs of atlas nuclei (i, j) for which nucleus j is among the K "closest" to i in terms of $\det D(i, j, R)$, or vice-versa i is among the K "closest" to j.

Inpainting. Nuclei that are "missing" in our 357-nuclei-atlas, mainly in the brain region, pose a challenge to the annotation problem: The atlas region posterior to the brain can freely match to target nuclei within the brain, taking the posterior body part with them. Therefore we *inpaint* the missing 201 nuclei into the training worms by taking one complete manual segmentation as reference and warping it to all the other training point clouds by means of Thin Plate Spline Warps. We inpaint the missing nuclei shapes by assigning the shape of the closest not-annotated nucleus in the respective manual segmentation.

2.2   Segmentation Hypotheses

For nuclei segmentation we use the Generalized Hough Transform (GHT) [10] with an ellipsoid as a template. We run GHT multiple times with a range of differently scaled and oriented templates. One GHT run yields one segmentation hypothesis per voxel, namly the template with highest *scoring* scale/orientation at the respective voxel position, where the score, $GHT(x) \in [0, 1]$, measures how

well the template fits the image gradient. From this abundant pool, per GHT run, we greedily select the n best-scoring hypotheses which do not overlap. Our final set of hypotheses is the union of sets of selected from the different GHT runs. Note that hypotheses from different GHT runs do, in general, overlap. This way we reduce the risk of missing nuclei. Figure 1 shows examples.

2.3   Active Graph Matching

Objective. Let n_T denote the number of segmentation hypotheses. An *assignment* $a_{i,j} \in \{0,1\}$ encodes whether atlas index $i \leq n_A$ is assigned to target index $j \leq n_T$. We denote the matrix of assignments as $\mathcal{A} := (a_{i,j})_{i=1,j=1}^{n_A,n_T}$. A bipartite matching is a matrix $\mathcal{A}$ which satisfies $\forall i \leq n_A : \sum_{j=1}^{n_T} a_{i,j} \leq 1$ and $\forall j \leq n_T : \sum_{i=1}^{n_A} a_{i,j} \leq 1$. I.e., an atlas nucleus can be matched to at most one target nucleus, and vice-versa. We define the energy of matching the atlas to the target with affine transformation $\mathbf{t}$, shape parameters $\mathbf{b}$, and matching $\mathcal{A}$, as

$$E(\mathcal{A}, \mathbf{b}, \mathbf{t}) := \sum_{i \leq n_A, k \leq n_T} \phi(i, k, \mathbf{b}, \mathbf{t}) \cdot a_{i,k} + \sum_{(i,j) \in \mathcal{N}_K, k, l \leq n_T} \psi(i, j, k, l, \mathbf{b}, \mathbf{t}) \cdot a_{i,k} \cdot a_{j,l} \tag{1}$$

where $\mathcal{N}_K$ is the neighborhood relation we defined on the atlas, cf. Section 2.1. *Unary potentials* $\phi(i, k, \mathbf{b}, \mathbf{t})$ encode the cost per assignment $a_{i,k}$. We define

$$\phi(i, k, \mathbf{b}, \mathbf{t}) := \lambda_1 \cdot \text{locDiff}(x_T(k), i, \mathbf{b}, \mathbf{t}) + \lambda_2 \cdot \text{shapeDiff}(s_T(k), i) + \lambda_3 \cdot \text{cost}(k) + c \tag{2}$$

where $x_T(k) \in \mathbf{R}^3$ is the center point of the k-th hypothesis, $k \leq n_T$, $s_T(k) \in \mathbf{R}^3$ is the target shape descriptor, $\text{cost}(k) := 1 - GHT(x_T(k))$ encodes how well the image data supports the k-th hypothesis, and c is a negative constant that serves as an incentive to make matches. Terms are weighted by positive constants λ. *Binary potentials* $\psi(i, j, k, l, \mathbf{b}, \mathbf{t})$ encode costs per pair of assignments, $a_{i,k}, a_{j,l}$:

$$\psi(i, j, k, l, \mathbf{b}, \mathbf{t}) := \lambda_4 \cdot \text{offsetDiff}(d_T(k, l), i, j, \mathbf{b}, \mathbf{t}) \tag{3}$$

where $d_T(k, l)$ denotes the offset between target nuclei k, l, namely $x_T(k) - x_T(l)$.

Optimization. To minimize (1) we first estimate initial parameters, $\mathbf{b}, \mathbf{t}$, and then alternate between minimization w.r.t. $\mathcal{A}$ (matching) and w.r.t. $\mathbf{b}, \mathbf{t}$.

Initial Atlas Parameters: We initialize the global shape parameters $\mathbf{b}$ to zero. As for an initial affine transformation $\mathbf{t}$, we align the first eigenvector of the point cloud given by all segmentation hypotheses with the anterior-posterior axis of the atlas such that the centers of gravity line up. We identify the correct rotation around this axis via the fact that nuclei are distributed asymmetrically along the dorso-ventral axis, while symmetrically along the left-right axis.

Optimal Matching: For fixed $\mathbf{b}, \mathbf{t}$ we minimize (1) w.r.t. the matching $\mathcal{A}$ with the Dual-Decomposition-based method of Torresani et al. [8]. In practice, considering all entries of $\mathcal{A}$ is intractable. Hence we only consider assignments $a_{i,k}$ for which $\text{locDiff}(x_T(k), i, \mathbf{b}, \mathbf{t})$ falls below a fixed threshold.

Table 1. Evaluation of annotation accuracy on 30 worms. Measures: median/mean(std), all in %. See text for description of scenarios (rows) and algorithms (columns). *Results presented as plot in [2], but numbers not given. **Results presented as plot in [2], but error measure not described and numbers not given.

	ActiveGM	**ActiveIGM**	ActiveHungarian	Long et al.
Synthetic	95/94(7)	-	93/88(12)	-
SemiAuto	92/90(8)	**93/92**(7)	79/77(9)	*/86(*)
Automatic	86/82(12)	**86/83**(11)	62/60(12)	**

Optimization of Atlas Parameters: For a fixed matching, the objective is the sum of squared residuals of an overdetermined system of equations which is linear in the global parameters $\mathbf{b}$ and $\mathbf{t}$, as described in the following. In the objective, only the terms locDiff and offsetDiff depend on atlas parameters. For locDiff each matched nucleus i entails three equations, namely

$$S \cdot R^{-1} x_T(k) - S \cdot \bar{x}_A(i) - S \cdot P(i)\mathbf{b} - S \cdot R^{-1}\mathbf{o} = (0,0,0)^T \qquad (4)$$

where S satisfies $S^T \cdot S = C(i,I)^{-1}$. Such an S exists in case $C(i,I)$ is symmetric and positive definite, which is the case in our practical setting. The equations are linear in the entries of R^{-1}, $\mathbf{b}$, and $\widetilde{\mathbf{o}} := R^{-1}\mathbf{o}$, respectively. For offsetDiff, each pair of matched neighbors i,j entails the following three equations:

$$G \cdot R^{-1}(x_T(k) - x_T(l)) - G \cdot (\bar{x}_A(i) - \bar{x}_A(j)) - G \cdot (P(i) - P(j))(b) = (0,0,0)^T \quad (5)$$

where G satisfies $G^T \cdot G = D(i)^{-1}$. Analogous to S, such a G exists in our practical setting. Overall we have, in practice, far more equations than parameters. Hence we can solve for optimal R, $\mathbf{o}$ and $\mathbf{b}$ with the method of least squares.

3   Results and Discussion

We run our method in a leave-one-out fashion on the 30 datasets used for atlas training (cf. Sec. 2). We consider three different scenarios. (1, *Synthetic*): We match the 357-nuclei atlas to the corresponding 357 target nuclei, which requires these 357 to be tagged in the manual segmentation. (2, *SemiAuto*): We match the atlas to all manually segmented target nuclei. (3, *Automatic*): We run fully automatic joint segmentation and annotation.

We run our algorithm in three different ways: (1, *ActiveGM*) with the 357-nuclei atlas, (2, *ActiveIGM*) with the inpainted 558-nuclei atlas, and (3, *ActiveHungarian*) without binary potentials. We run ActiveIGM only for the real-world scenarios. We run ActiveHungarian without in-painting in the synthetic scenario, and with in-painting in the real-world scenarios.

As for the parameters of our method, we always set: $\lambda_2 := 1$, $c := -150$, $K := 6$, and as optimization steps 3 times the sequence $\mathcal{A}, \mathbf{t}$, followed by 3 times $\mathcal{A}, \mathbf{b}$. We consider the first two modes of variation in $\mathbf{b}$. As for locDiff we set $\lambda_1 := 0$ for ActiveGM, and $\lambda_1 := 1$ for ActiveHungarian. As for offsetDiff we
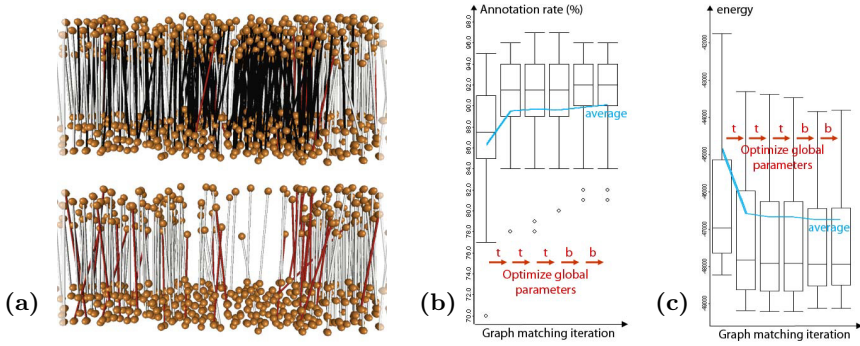
Fig. 3. (a) Close-up to matching results in the head of an exemplary worm. Top: inpainted atlas; bottom: partial atlas. Inpainting leads to better matching performance. White lines: correct annotation; black lines: inpainted nuclei, no ground truth available; red lines: annotation errors (fewer on top). **(b)** Evolution of annotation accuracy for semi-automatic matching scenario. X-axis: matching iteration. Y-axis: fraction of correctly annotated nuclei. **(c)** Evolution of the respective matching energy.

set $\lambda_4 := 1$ for ActiveGM, but $\lambda_4 := 0$ for ActiveHungarian since it cannot handle binary potentials. As target cost weight we set $\lambda_3 := 0$ in all but the fully automatic scenario, where we set $\lambda_3 := 10$. All parameter values were chosen heuristically.

To measure annotation accuracy, in case of manual segmentations, we count the fraction of correctly annotated nuclei. For the fully automatic scenario, we count the fraction of matched segmentation hypotheses whose center points lie within the respective ground truth nucleus, or are at most one average nucleus radius apart from the true center point. Table 1 lists the results for all experiments described above. For reference we also include the result of Long et al. [2].

Apart from annotation accuracy we analyzed the optimality of matching in terms of the gap between lower bound and found energy: In the synthetic and the semi-automatic scenario, lower bounds are tight, i.e. here we find the globally optimal matching. As for the fully autkmatic scenario, the matching problem is solved approximately with an average duality gap of about $2c$.

Discussion. For the task of annotating manual segmentations of nuclei our average annotation rate of 92% considerably outperforms the result of Long et al. who report an average of 86%. Note that furthermore Long et al. need an additional image channel which our method does not. For the fully automatic task our median/average annotation rate of 86/83% approaches the rate that Long et al. achieved in the much simpler partly manual scenario.

Employing 2nd order graph matching instead of just the Hungarian algorithm makes a huge difference: ActiveHungarian works relatively well *only* in the synthetic scenario, while the inferiority as compared to ActiveGM increases as the matching problem gets more sophisticated: In the order of complexity of the matching problem (top to bottom in Table 1), ActiveGM is on average 6%, 15%, and 23% better than ActiveHungarian, respectively.

Figure 3(a) shows the benefit of using the in-painted atlas instead of the 357-nuclei-atlas. Figure 3(b,c) shows how the annotation rate and the respective value of the objective evolve during Active Graph Matching iterations.

Neglecting location differences in the 2nd order energy, i.e. $l_1 := 0$ instead of $l_1 := 1$ for ActiveGM, yields considerably better annotation accuracy. We argue that this is due to the respective much more flexible local deformation model. Note that locDiff $= 0$ means that the objective is invariant w.r.t. **o**. However, in practice we still need **o** for selecting the assignments we consider in the matching problem (cf. Section 2.3), hence we always derive it via locDiff.

Conclusion. We have presented *active graph matching*, a method that combines active shape models with graph matching in one objective and provides an approach for global optimization. With this method we do not only outperform the current state of the art in annotating manual segmentations of nuclei in *C. elegans* L1 larvae, but furthermore define the state of the art in solving both segmentation and annotation simultaneously in a fully automatic fashion. We hypothesize that our method will be highly beneficial for the equally relevant task of nuclei annotation in later stages of *C. elegans* development, where nuclei are more numerous and more densely packed.

References

1. Sarov, M., et al.: A Genome-Scale Resource for In Vivo Tag-Based Protein Function Exploration in C. Elegans. Cell 150(4), 855–866 (2012)
2. Long, F., Peng, H., Liu, X., Kim, S.K., Myers, E.: A 3D Digital Atlas of C. Elegans and its Application to Single-Cell Analyses. Nature Methods 6, 667–672 (2009)
3. Aerni, S.J., Liu, X., Do, C.B., Gross, S.S., Nguyen, A., Guo, S.D., Long, F., Peng, H., Kim, S.S., Batzoglou, S.: Automated Cellular Annotation for High-resolution Images of Adult C. Elegans. Bioinformatics 29(13), i18–i26 (2013)
4. Qu, L., Long, F., Liu, X., Kim, S., Myers, E., Peng, H.: Simultaneous Recognition and Segmentation of Cells. Bioinformatics 27(20), 2895–2902 (2011)
5. Riklin Raviv, T., Ljosa, V., Conery, A.L., Ausubel, F.M., Carpenter, A.E., Golland, P., Wählby, C.: Morphology-guided graph search for untangling objects: C. elegans analysis. In: Jiang, T., Navab, N., Pluim, J.P.W., Viergever, M.A. (eds.) MICCAI 2010, Part III. LNCS, vol. 6363, pp. 634–641. Springer, Heidelberg (2010)
6. Peng, H., Long, F., Liu, X., Kim, S.K., Myers, E.: Straightening Caenorhabditis Elegans Images. Bioinformatics 24(2), 234–242 (2008)
7. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active Shape Models - Their Training and Application. CVIU 61(1), 38–59 (1995)
8. Torresani, L., Kolmogorov, V., Rother, C.: A dual decomposition approach to feature correspondence. IEEE TPAMI 35(2), 259–271 (2013)
9. Zhou, F., De La Torre, F.: Deformable graph matching. In: CVPR, pp. 2922–2929 (2013)
10. Khoshelham, K.: Extending Generalized Hough Transform to Detect 3D Objects in Laser Range Data. In: ISPRS Workshop Laser Scanning, p. 206 (2007)