Discriminative Detection and Alignment in Volumetric Data

Dominic Mai[1,2], Philipp Fischer[1], Thomas Blein[4], Jasmin Dürr[3],
Klaus Palme[2,3], Thomas Brox[1,2], and Olaf Ronneberger[1,2]

[1] Lehrstuhl für Mustererkennung und Bildverabeitung, Institut für Informatik
[2] BIOSS Centre of Biological Signalling Studies
[3] Institut für Biologie II, Albert-Ludwigs-Universität Freiburg
[4] INRA Versailles
maid@informatik.uni-freiburg.de

Abstract. In this paper, we aim for detection and segmentation of *Arabidopsis thaliana* cells in volumetric image data. To this end, we cluster the training samples by their size and aspect ratio and learn a detector and a shape model for each cluster. While the detector yields good cell hypotheses, additionally aligning the shape model to the image allows to better localize the detections and to reconstruct the cells in case of low quality input data. We show that due to the more accurate localization, the alignment also improves the detection performance.

1 Introduction

In biomedical image analysis, one is often interested in explaining and measuring observed image structures by fitting a model to the data. This enables a statistical analysis of imaged data, e.g., gene expression patterns. At the same time, it can be interesting to highlight deviations from the standard model, i.e., those patterns that are not explained by the fitted model.

In this paper, we focus on segmentation of specific cell types in volumetric data. A standard approach is to run a segmentation method, e.g. watershed segmentation, followed by a classification of the obtained regions [9]. This corresponds to approaches on object class segmentation that run classifiers on region hypotheses [3].

The opposite approach is to first run a discriminative detector of a trained model on the image data to obtain object class hypotheses, and then to obtain a refined segmentation by aligning the detected model to the image data [2].

In this paper, we focus on the latter approach and extend the idea to volumetric cell data. To this end, we set up a volumetric variant of HOG descriptors [5] inspired by [8] who construct a 3D HOG descriptor for 2D + time data for action recognition. We cluster the set of training samples and set up a cell model for each cluster. The model consists of a linear SVM for detection in a sliding window fashion, and a shape template for alignment. Compared to [2], we improve on the model template used for alignment. In particular, we learn a common mean

J. Weickert, M. Hein, and B. Schiele (Eds.): GCPR 2013, LNCS 8142, pp. 205–214, 2013.

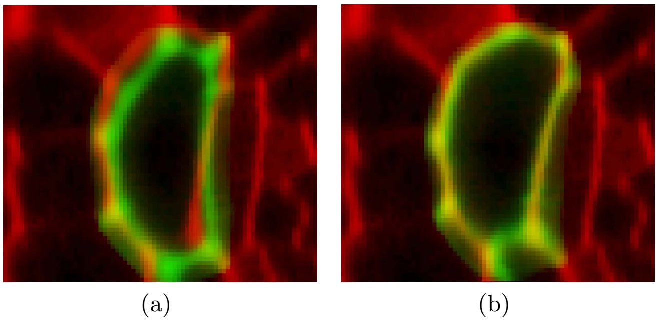


Fig. 1. The two images show a slice of the root in red. The green overlays show a detected cell without (a) and with alignment (b).

template from the training samples [13]. In this approach, a diffeomorphism is estimated between the initial template and all positive training samples, and the mean template is optimized with regard to the total deformation of all samples. For the alignment we use state-of-the-art elastic registration methods [1,11].

We quantitatively evaluate the approach by detecting and reconstructing cells of *Arabidopsis thaliana* in dense volumetric tissue. *Arabidopsis thaliana* is a model organism that is widely used in the research on plant development and genetics [7,10]. The structure of the root is visualized in Fig. 2(a). We demonstrate that the alignment improves the accuracy of the cell segmentation. Moreover, the alignment improves detection performance since the detections are better localized after alignment; see Fig. 1.

The detection based approach also allows to obtain good segmentations on lower quality data when a segmentation guided approach, such as in [9] yields unsatisfactory results.

2 Training the 3D Cell Detectors

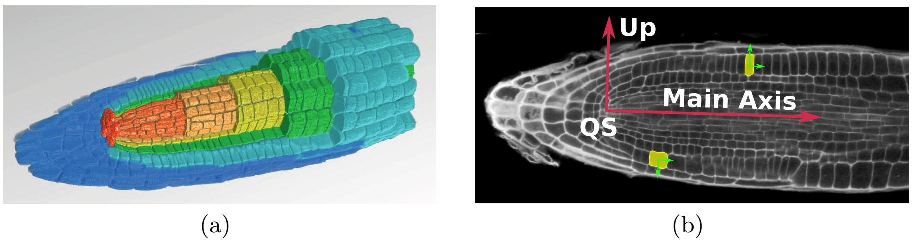
In this section we will describe the training of the discriminative detection filters and the features that we use. The training input is a volumetric 3D image[1] $I : \Omega \to \mathbb{R}$; $\Omega \subset \mathbb{R}^3$ of the root, with segmentation masks $S_i : \Omega \to \{0, 1\}$ for the cells that will be our positive examples.

In a classical detection framework all samples are assumed to have the same orientation, in our setting however, we have a rotational axis in the center of the root and the sliding window has to be rotated around this axis. Therefore, in this special case we determine a *root coordinate system* for rotation normalization. The fitting of a coordinate system to the root can be done automatically as shown in [12]. We will, however, use the simplified version illustrated in Fig. 2(b).

The outline of the training algorithm is as follows:

[1] In the biomedical field, the term “image” is used also for volumetric images.

1. Extract orientation normalized positive training images of cells by using the segmentation masks and the root coordinate system
2. Perform a k-means clustering using the bounding boxes of the shapes
3. Create an average segmentation mask for every cluster
4. Randomly sample orientation normalized negative images for every cluster
5. Train a linear SVM for every cluster using 3D HOG features



(a) (b)

Fig. 2. In (a) the structure of the root can be clearly seen: It consists of cells of different shapes and sizes organized in concentric cylindrical layers. We use a cylindrical root coordinate system (b) to normalize the orientation of our training images. It is defined by the position of the *quiescent center* (QS), the translational direction along the main axis and a radial up-vector component pointing outwards from this axis.

2.1 Data Normalization and Clustering

The data that we are dealing with has rotational symmetry along one axis. The traditional sliding window approach does not deal with rotations. Therefore, we normalize them out for the training and the detection. This is similar to the different scales that one usually encounters in natural images. For the training they are normalized out and a filter is trained for one scale only. During test time, the filter is then applied to an image pyramid containing different scales.

The root coordinate system consists of the location and orientation of the main axis and an arbitrary but fixed component defining the “up” direction perpendicular to this axis. We use it to attach an orientation to every segmentation mask S_i and to compute a rigid transformation $\mathbf{H}_i : \mathbb{R}^3 \to \mathbb{R}^3$ that normalizes for location and rotation. The location normalization maps the center of mass of a cell to the coordinate origin. The resulting orientation normalized segmentation masks are then $\hat{S}_i = S_i \circ \mathbf{H}_i$, where $\circ$ denotes function composition, i.e. $(S_i \circ \mathbf{H}_i)(\mathbf{x}) = S_i(\mathbf{H}_i(\mathbf{x}))$

After normalization we perform a k-means clustering on the shape of the cuboidal bounding boxes of $\hat{S}_i$. They are defined as $\mathbf{s}_i = \mathbf{b}_{\text{high}}(\hat{S}_i) - \mathbf{b}_{\text{low}}(\hat{S}_i)$, with $\mathbf{b}_{\text{high}}, \mathbf{b}_{\text{low}}$ being functions that extract the tight bounding box, i.e. $\mathbf{b}_{\text{high}} : [\Omega \to \{0,1\}] \to \mathbb{R}^3$, and $\mathbf{b}_{\text{low}}$ accordingly. For each of the K clusters we define a set $\mathcal{C}_k = \{c_1, \ldots, c_{N_k}\}$ containing the indices of the segmentation masks. We use the absolute bounding box sizes for clustering (instead of the common aspect ratios), because we know the scale of the microscopic images. In the experiments section, we evaluate the effect of the parameter K.

After the clustering, we determine the ranges $\mathbf{R}_{\text{high}}, \mathbf{R}_{\text{low}}$ of the filter box such that it can enclose all the shapes in a cluster.

$$\mathbf{R}^k_{\text{low}} = \mathbf{b}_{\text{low}}(\bigcup_{i \in \mathcal{C}_k} \hat{S}_i) - \boldsymbol{\delta}, \quad \mathbf{R}^k_{\text{high}} = \mathbf{b}_{\text{high}}(\bigcup_{i \in \mathcal{C}_k} \hat{S}_i) + \boldsymbol{\delta} \tag{1}$$

The parameter $\boldsymbol{\delta} \in \mathbb{R}^3$ specifies how much "context" we want to add to either side of the training images. As shown in [5], context improves the detector's performance significantly. The support domain of a filter then becomes $\Omega_k = \{\mathbf{x} \in \mathbb{R}^3 | \mathbf{R}^k_{\text{low}} \leq \mathbf{x} \leq \mathbf{R}^k_{\text{high}}\}$ where we define the "lesser than" relations for vectors, such that the relations for all components must be true. The positive training images X_i are obtained as $\hat{X}_i = I \circ \mathbf{H}_i$, defined on Ω_k for the respective cluster.

We create an average segmentation mask for the detection filter that gives us an estimate of the most likely location of a detected cell. Following [2] we do this by taking the 0.5 level set of the averaged normalized segmentation masks.

$$S^k(\mathbf{x}) = \begin{cases} 1, \text{if} \quad \frac{1}{|\mathcal{C}_k|} \sum_{i \in \mathcal{C}_k} \hat{S}_i(\mathbf{x}) \geq 0.5 \\ 0, \text{otherwise} \end{cases} \tag{2}$$

The negative examples are obtained by randomly sampling rotation normalized boxes from the training root. We make sure not to accidentally include positive training examples. In the next sections we will present the features that we use to represent our images.

2.2 Histograms of Oriented Gradients

The majority of information in an image lies in its edges and HOG has proven well for detection in 2D pictures. In our framework we build upon this idea and create a 3D detector in a similar manner.

We define a direction histogram binning function $b_i : \mathbb{R}^3 \to \mathbb{R}$, using N_{d} directions $\{\mathbf{d}_1, \ldots, \mathbf{d}_{N_{\text{d}}}\}$ with $\mathbf{d}_i \in \mathbb{R}^3$ and $\|\mathbf{d}_i\| = 1$ equally distributed on the unit sphere (here we use the vertices of a dodecahedron).

$$b_i(\mathbf{n}) = e^{-\frac{1}{2}\left(\frac{\mathbf{d}_i^\top \frac{\mathbf{n}}{\|\mathbf{n}\|} - 1}{\sigma}\right)^2} \tag{3}$$

where σ is selected such that the weight is approx. 0.5 at the middle between two bins, i.e. if $\mathbf{d}_j$ is a direct neighbor of $\mathbf{d}_i$, then $b_i((\mathbf{d}_i + \mathbf{d}_j)/2) \approx 0.5$. For easier notation, all binning functions are combined into one vector-valued binning function $\mathbf{b} : \mathbb{R}^3 \to \mathbb{R}^{N_{\text{d}}} \quad \mathbf{b}(\mathbf{n}) = [b_1(\mathbf{n}), \ldots, b_{N_{\text{d}}}(\mathbf{n})]^\top$.

Mapping an image $I : \Omega \to \mathbb{R}$ to and oriented gradient map $\mathbf{O} : \Omega \to \mathbb{R}^{N_{\text{d}}}$ is then defined by the mapping $\mathbf{G} : [\Omega \to \mathbb{R}] \to [\Omega \to \mathbb{R}^{N_{\text{d}}}]$

$$\mathbf{G}(I)(\mathbf{x}) := \|\nabla I(\mathbf{x})\| \cdot \frac{\mathbf{b}(\nabla I(\mathbf{x}))}{\|\mathbf{b}(\nabla I(\mathbf{x}))\|} \tag{4}$$

2.3 Combining and Subsampling

To add robustness to small local deformations, in the 2D HOG framework, the gradients are accumulated in feature cells. We adopt this procedure and convolve the feature map $\mathbf{G}(I)$ with a triangular filter kernel $T : \mathbb{R}^3 \rightarrow \mathbb{R}$ with radius w,

$$T(\mathbf{x}) = \begin{cases} 1 - \frac{\|\mathbf{x}\|_{\mathrm{L}_1}}{w} & \text{, if } \|\mathbf{x}\|_{\mathrm{L}_1} < w \\ 0 & \text{else} \end{cases} \tag{5}$$

This is equivalent to a soft binning in the standard setting. We normalize $\mathbf{G}(I)$ as $\tilde{\mathbf{G}}(I) = (T * \mathbf{G}(I))/(\left\|T * \mathbf{G}(I) + \epsilon\right\|)$ to have unit length at every position $\mathbf{x}$, while assuring not to include locations with too little structure by regularizing it with a suitable ϵ. To obtain the feature vector used for the SVM training we now discretize $\tilde{\mathbf{G}}(I)$ by sampling it at regular grid positions $\mathbf{g}_i \in \Omega$ for $i \in \{1, \ldots, N_{\mathrm{g}}\}$ and concatenate the vectors to a single feature vector $\mathbf{F}(I) = \left[\tilde{\mathbf{G}}(I)(\mathbf{g}_1); \ldots; \tilde{\mathbf{G}}(I)(\mathbf{g}_{N_{\mathrm{g}}})\right]$, where $\mathbf{F}(I) \in \mathbb{R}^{N_{\mathrm{d}} \cdot N_{\mathrm{g}}}$

2.4 SVM Training

We train a linear SVM [4] for every cluster $\mathcal{C}_k$. The amount of sampled negatives is five times the amount of positives for every cluster. As the dimensionality of our feature vector is orders of magnitude bigger than the number of examples per training (roughly 500.000 : 500), we precompute the kernel matrix $\mathbf{X}^T\mathbf{X}$, where $\mathbf{X}$ contains the feature vectors of the positive and negative examples as column vectors.

3 Sharp Mean Image Generation

For a classical sliding window detector, the localization hypothesis for a detection is the bounding box of the filter. One can improve on this localization by incorporating the segmentation masks from the training. E.g. by using the average segmentation mask S_k (Eqn. 2). We propose to use elastic alignment of a *sharp mean image* to further improve the localization.

In this section we describe how we create a sharp mean image Z_k for each cluster $\mathcal{C}_k$. For an illustration see Fig. 3(b). For the generation we follow the ideas from [13], i.e. we want a sharp average image with respect to the intensities and deformations present in the cluster $\mathcal{C}_k$. To this end we need a reference image, to which we align all other images from the cluster. As reference image we take the image X_{r_k} with $r_k \in \mathcal{C}_k$, whose bounding box size has the minimal distance to the average bounding box. We use the publicly available diffeomorphic registration from the ANTS[1] toolkit to compute the transformations $\theta_{m,f} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that maps the image I_m onto the image I_f. We use normalized cross correlation as data term with a diffeomorphic regularizer. For the computation of the sharp mean image $Z_k : \Omega \rightarrow \mathbb{R}$, we use the transformations $\boldsymbol{\theta}_{j,r_k}$ to normalize the

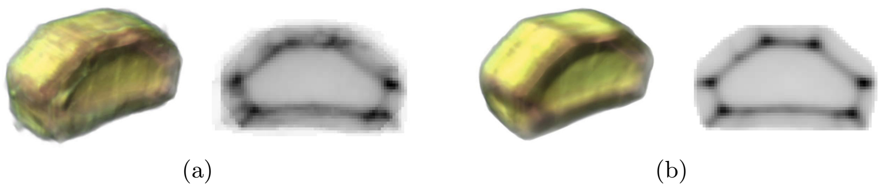


Fig. 3. Volume rendering and slice view of an average cell formed from all positive examples of a cluster. In (a) the intensities of the rigidly aligned cells are averaged. The *sharp mean image* (b) resembles the structure much cleaner.

shape of every training image X_j in the cluster to the reference shape X_{r_k}. Then we average the intensities and use the average inverse transformation to compute the sharp mean image. Note that a relative deformation field can be computed from the absolute transformation by subtracting the identity.

$$Z_k = \underbrace{\left(\frac{1}{|\mathcal{C}_k|}\sum_{j\in\mathcal{C}_k} X_j \circ \boldsymbol{\theta}_{j,r_k}\right)}_{\text{average intensities}} \circ \underbrace{\left(\mathrm{Id} + \frac{1}{|\mathcal{C}_k|}\sum_{j\in\mathcal{C}_k}\left(\boldsymbol{\theta}_{j,r_k}^{-1} - \mathrm{Id}\right)\right)}_{\text{average deformation}} \tag{6}$$

For the generation of Z_k we dilate the segmentation masks S_i used to cut out the images X_i, as the segmentation only captures half of the cell wall and a little extra structure around the cell is needed for the elastic registration's data term to work properly. The "tight" sharp segmentation masks S_{Z_k} which will be used in the alignment, are computed by averaging the individual segmentation masks $\hat{S}_j$ after applying the same transformations as for the images.

4 Detection and Alignment

For the detection we apply a sliding window approach to the original data set and rotated versions of it, denoted as B_i. The rotations are sampled in 10 degree steps around the axis, defined by the root coordinate system, resulting in 36 data sets. For every B_i we compute the score maps for every detection filter at the grid positions. The necessary convolutions are computed efficiently in the Fourier domain. We ensure that within the volume of the average segmentation masks S^k (Eqn. 2), at most one local maximum appears. The lower scoring hypotheses are suppressed. Finally, we perform an upsampling by cubic interpolation of the score maps to obtain the detection locations with higher accuracy. As a last step we transform the detection locations back to the original root and assemble them in a greedy fashion, by starting with the highest scoring detection hypothesis.

4.1 Alignment

We perform an alignment for every detection. As initialization for the alignment we put the sharp mean image Z_k at the location where its tight segmentation mask S_{Z_k} has the biggest overlap with the average segmentation mask S_k

from the detector (Fig. 1(a)). We use a fast combinatorial registration from the ViBE-Z [11] toolkit, where we replaced the data term (originally a locally normalized cross correlation) by a nonlinear gradient direction and gradient magnitude comparison, to obtain the same robustness properties as a HOG descriptor. The similarity of a local region Ω_i of the fixed image I and transformed moving image J is defined as

$$E_{\mathrm{sim}}(I,J) = \frac{\int_{\Omega_i} \left\|\nabla I\right\| \cdot \left\|\nabla J\right\| \cdot \exp\left(-\frac{1}{2\sigma^2}\left(\cos^{-1}\left(\frac{(\nabla I)^\top \nabla J}{\|\nabla I\|\|\nabla J\|}\right)\right)^2\right) d\mathbf{x}}{\sqrt{\int_{\Omega_i} \left\|\nabla I\right\|^2 d\mathbf{x}} \cdot \sqrt{\int_{\Omega_i} \left\|\nabla J\right\|^2 d\mathbf{x}}} \tag{7}$$

We opted to choose this registration over the ANTS registration used in training as we do not need a diffeomorphic mapping and the runtime is faster by several orders of magnitude.

The ViBE-Z registration has several parameters that need to be adapted to the structures present in the data: The grid spacing α, the patch radius β, the label radius γ, and the Elasticity λ. We optimize these parameters automatically on the training set by maximizing the *intersection over union* (IOU) between the transformed sharp mean segmentation mask and the ground truth segmentation of a detection. For two volumes V and W the IOU is defined as

$$M_{iou}(V,W) = \frac{|V \cap W|}{|V \cup W|} \tag{8}$$

In the following section we provide quantitative evidence that it is crucial to work with the correct parameter settings.

5 Experiments

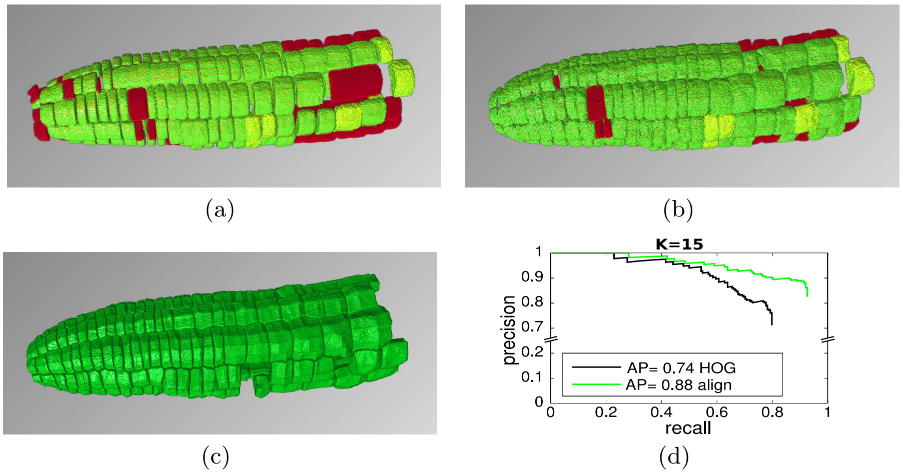
For the quantitative evaluation we had two roots with ground truth segmentations available. Ground truth segmentations are obtained by manually verifying a watershed segmentation computed on enhanced data [9]. This is very cumbersome work, as a root contains roughly 2.500 cells. During the verification, wrongly segmented cells are marked as invalid. In this study we concentrate on the about 250 cells of the cortex layer that have the most complex shape: The Green layer in Fig. 2(a).

We measure the quality of a detection hypothesis using the IOU (Eqn. 8) of its segmentation mask with the ground truth segmentation. For aligned detections, we take the aligned segmentation mask of the sharp mean image. For the precision-recall diagrams we apply the PASCAL VOC [6] criterion: A detection is accepted as a true positive iff $M_{\mathrm{iou}} \geq 0.5$. All lower ranked detections in this location count as false positives. Note that in contrast to the PASCAL VOC challenge for natural images, we use the GT segmentations and not only the bounding boxes which makes the task more demanding. The average precision is computed as the area under the precision-recall curve.

Table 1. Effect of the number of clusteres K on the average precision (AP) with and without alignment

K	2	5	10	15	25
AP	0.45	0.58	0.63	**0.74**	0.72
AP-aligned	0.58	0.74	0.75	**0.88**	0.86

The test root that we used for evaluation has an image size of $944 \times 413 \times 360$ Voxel. We sampled $3 \times 36 = 108$ overlapping rotation normalized boxes of the size $301 \times 101 \times 131$ around the main axis of the root coordinate system. We found that it is sufficient to sample in 10° steps. On a six-core machine the detection for $K = 15$ takes about 1 hour while the majority of the time is spent on the computation of the HOG feature for each rotation normalized box. The remapping of the detections to the original space and the alignment takes another 30 minutes. Notice that the detection and alignment steps are nearly perfectly parallelizable – we noticed a quasi linear scaling of the processing time when we used more computers in parallel. For the alignment, we found that a parameter setting of $\alpha = 4, \beta = 7, \gamma = 10$, and $\lambda = 5$ gives best results on the training data. Table 1 and Fig. 4 summarize our findings: The choice of the right amount of clusters is crucial for the overall performance of our system. While a too small number of clusters cannot capture the variance present in the data, a too big number does not leave enough positive training examples for the SVM. The alignment of the sharp mean image boosts the AP by approximately +14%, independent of



(a) (b)

(c) (d)

Fig. 4. Rendering of the best detection result ($K = 15$) before alignment (a) and after alignment (b). The ground truth is shown in (c). *Red* cells have an IOU below 0.5, *yellow* cells are detections at locations where GT segmentations are missing and *green* cells are correctly detected. In (d) you can see the corresponding *precision-recall* graphs. Notice how the alignment improves not only the AP but the overall quality of the reconstruction.

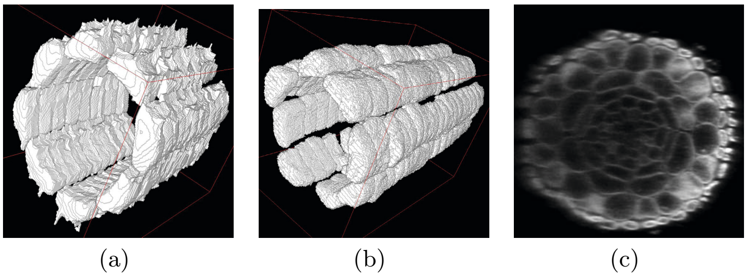


Fig. 5. Qualitative example on data of low recording quality. The watershed segmentation (a) leaks out at some cell-borders. Our approach (b) produces a more naturally looking result. A slice of the original raw data can be seen in (c).

the number of clusters. For the true positive detections, the average localization accuracy gets a boost from $\varnothing IOU = 0.66$ to $\varnothing IOU = 0.78$. When we use the average image instead of the sharp mean image(Fig. 3(a)), the AP gets reduced by about 2%. The alignment is also the reason for the gain in AP: Some of the detections that were poorly localized before the alignment get a boost in IOU big enough to make them count as true positive detections afterwards. Notice that we do not get any false positive detections outside the sought layer. False positive detections occur either when one GT cell is falsely explained by two detections or vice versa.

In Fig. 5 you can see a qualitative illustration of the ability of our approach to generalize to data of worse recording quality. While the watershed segmentation fails at many of the cell borders, we achieve a visually much more plausible result.

6 Conclusions

In this work we presented and evaluated how an elastic alignment can improve detection in volumetric data. With the *alignment* of a sharp mean image to the data we were able to improve the localization accuracy and average precision substantially. While the general idea of aligning some template to the data should also hold in 2D, the realization would be more difficult: One would have to deal with occlusions, viewpoint dependent distortions of the image and the articulated nature of some object classes, like e.g. humans.

However, we believe that this work is a significant step towards the goal to explain every voxel in a newly recorded data set based on learned Models.

Acknowledgements. This work was supported by the Excellence Initiative of the German Federal and State Governments: BIOSS Centre for Biological Signalling Studies (EXC 294) and the Bundesministerium für Bildung und Forschung (SYSTEC, 0101-31P5914). Philipp Fischer is supported by the Deutsche Telekom Stiftung.

References

1. Avants, B.B., Epstein, C.L., Grossman, M., Gee, J.C.: Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. Medical Image Analysis 12(1), 26–41 (2008)
2. Brox, T., Bourdev, L., Maji, S., Malik, J.: Object segmentation by alignment of poselet activations to image contours. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) (2011)
3. Carreira, J., Sminchisescu, C.: CPMC: Automatic Object Segmentation Using Constrained Parametric Min-Cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence (2012)
4. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Schmid, C., Soatto, S., Tomasi, C. (eds.) International Conference on Computer Vision & Pattern Recognition (2005)
6. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision 88(2), 303–338 (2010)
7. Fernandez, R., Das, P., Mirabet, V., Moscardi, E., Traas, J., Verdeil, J., Malandain, G., Godin, C.: Imaging plant growth in 4d: robust tissue reconstruction and lineaging at cell resolution. Nature Methods 7(7), 547–553 (2010)
8. Kläser, A., Marszalek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: BMVC (2008)
9. Liu, K., Schmidt, T., Blein, T., Dürr, J., Palme, K., Ronneberger, O.: Joint 3d cell segmentation and classification in the arabidopsis root using energy minimization and shape priors. In: IEEE International Symposium on Biomedical Imaging (ISBI) (2013)
10. Marcuzzo, M., Quelhas, P., Campilho, A., Maria Mendonça, A., Campilho, A.: Automated arabidopsis plant root cell segmentation based on svm classification and region merging. Comput. Biol. Med. 39(9), 785–793 (2009)
11. Ronneberger, O., Liu, K., Rath, M., Ruess, D., Mueller, T., Skibbe, H., Drayer, B., Schmidt, T., Filippi, A., Nitschke, R., Brox, T., Burkhardt, H., Driever, W.: Vibe-z: A framework for 3d virtual colocalization analysis in zebrafish larval brains. Nature Methods 9(7), 735–742 (2012)
12. Schmidt, T., Keuper, M., Pasternak, T., Palme, K., Ronneberger, O.: Modeling of sparsely sampled tubular surfaces using coupled curves. In: Pinz, A., Pock, T., Bischof, H., Leberl, F. (eds.) DAGM and OAGM 2012. LNCS, vol. 7476, pp. 83–92. Springer, Heidelberg (2012)
13. Wu, G., Jia, H., Wang, Q., Shen, D.: Sharpmean: Groupwise registration guided by sharp mean image and tree-based registration. NeuroImage 56(4) (2011)