A probabilistic model combining deep learning and multi-atlas segmentation for semi-automated labelling of histology

Alessia Atzeni[1], Marnix Jansen[2], Sebastien Ourselin[3*], and J. Eugenio Iglesias[1]

[1] Medical Physics and Biomedical Engineering, University College London, UK
[2] University College London Hospital, London, UK
[3] Wellcome EPSRC Centre for Interventional and Surgical Sciences (WEISS), University College London, UK

Abstract. Thanks to their high resolution and contrast enhanced by different stains, histological images are becoming increasingly widespread in atlas construction. Building atlases with histology requires manual delineation of a set of regions of interest on a large amount of sections. This process is tedious, time-consuming, and rather inefficient due to the high similarity of adjacent sections. Here we propose a probabilistic model for semi-automated segmentation of stacks of histological sections, in which the user manually labels a sparse set of sections (e.g., one every n), and lets the algorithm complete the segmentation for other sections automatically. The proposed model integrates in a principled manner two families of segmentation techniques that have been very successful in brain imaging: multi-atlas segmentation (MAS) and convolutional neural networks (CNNs). Within this model, we derive a Generalised Expectation Maximisation algorithm to compute the most likely segmentation. Experiments on the Allen dataset show that the model successfully combines the strengths of both techniques (effective label propagation of MAS, and robustness to misregistration of CNNs), and produces significantly more accurate results than using either of them independently.

1 Introduction

Histological sections, which can be digitised at sub-micron resolution, allow to differentiate and characterise brain substructures that are not visible with mm-scale imaging (e.g., MRI), and are becoming increasingly popular for building high resolution brain atlases, e.g., BigBrain [1] or Allen [2]. An important component of many of these atlases is a set of associated manual delineations of regions of interest. Manual segmentation is however tedious and time-consuming – and thus expensive. In histological datasets, where stacks of 2D sections are labelled to create a 3D segmentation, manually delineating adjacent sections is very inefficient due to their similarity. A possible solution is the use of semi-automated algorithms, which allow labelling one slice every n, letting the method complete the segmentation task automatically, with the possibility of final user refinement.

*SO is currently with the School Biomed. Eng. Imag. Sci., King's College London

Many semi-automated algorithms rely on the introduction of user defined scribbles or boundary points, which are treated as prior information by the algorithm to produce a dense segmentation of the whole image. If the computational complexity of the method is low enough, the user can interactively review the output and add or remove scribbles/points to correct mistakes, refining the segmentation until it is satisfactory. Popular semi-automated segmentation techniques include Random walker [3] or GeoS [4]. For 3D modalities like MR or CT, one can label a subset of slices and use them as input for these algorithms to complete the segmentation for the whole volume. However, for stacks of histological images, these techniques cannot be used due to the absence of 3D continuity between sections.

An alternative approach is to treat the labelled sections as training data, and use supervised segmentation techniques to segment the unlabelled sections in between. A very successful family of techniques in brain image segmentation are multi-atlas based [5, 6]. Multi-atlas segmentation (MAS) relies on non-rigid registration between a set of atlases and a test image. The deformations resulting from the registration are used to propagate the atlas labels to the novel image coordinates, where the segmentation of each pixel is decided through a label fusion approach. These techniques are well suited for inter-slice labelling as long as the registered sections are not too far apart, such that the registration can be expected to be good.

Meanwhile, deep learning techniques, best represented by convolutional neural networks (CCNs), have become increasingly popular in medical image segmentation. Deep learning can be directly applied to semi-automated segmentation of medical images. For example, a 3D U-net was trained on few manually annotated orthogonal slices in [7], in order to produce a segmentation for the whole volume. The negative effects of the limited training data were ameliorated with aggressive data augmentation.

The present paper integrates deep learning and label fusion into a joint probabilistic model in a principled way. Along with the model, we present an inference method – based on the Generalised Expectation Maximisation (GEM) algorithm – to compute the most likely segmentation for an input histological image, given the labelled neighbouring sections. The proposed algorithm successfully combines the advantages of the two techniques, inheriting: 1. from CNNs, the robustness to registration errors, which might happen due to artefacts or large separation between the sections to register; and 2. from MAS, the ability to preserve anatomical shape, including faint or invisible boundaries that rely on prior knowledge, e.g., between brain substructures or cortical regions.

2 Methods

2.1 Probabilistic model

The graphical model of the proposed probabilistic framework is shown in Fig. 1. Let $\{I_n(\boldsymbol{x})\}_{n=1,\dots,N}$ be N histological sections defined on discrete coordinates $\boldsymbol{x}$

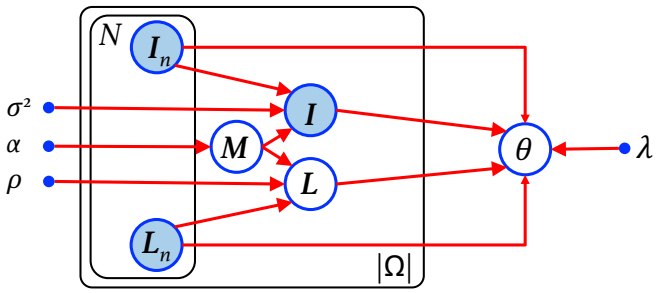

Fig. 1. Graphical model representing the relationship between the model variables. Replications are illustrated with plates. Shaded variables are observed.

over an image domain $\Omega \subset \mathbb{R}^2$, for which a manual segmentation is available. In a similar manner, let $\{L_n(\boldsymbol{x})\}_{n=1,\ldots,N}$ be the corresponding (manual) segmentations. $\{I_n(\boldsymbol{x})\}$ and $\{L_n(\boldsymbol{x})\}$ define thus a training dataset of atlases. We assume that these atlases have been pre-registered to a test image $I(\boldsymbol{x})$, whose labels $L(\boldsymbol{x})$ are unknown. A label fusion approach aims to estimate the label map $\boldsymbol{L}$ associated with $\boldsymbol{I}$, given the registered atlases. Here we assume the availability of a probabilistic fusion algorithm that produces a posterior probability of the segmentation p^f that factorises over voxels:

$$p^f(\boldsymbol{L}|\{\boldsymbol{I}_n\},\{\boldsymbol{L}_n\},\boldsymbol{I}) = \prod_{\boldsymbol{x}\in\Omega} p^f_{\boldsymbol{x}}(L(\boldsymbol{x})|\ \{\boldsymbol{I}_n\},\{\boldsymbol{L}_n\},\boldsymbol{I}).$$

Let $\boldsymbol{\theta}$ be the parameters of a semantic segmentation neural network trained on all images and corresponding segmentations within our framework – both the atlases and the test image. We can then derive a discriminative probability density function on $\boldsymbol{\theta}$, conditioned on the training data:

$$\begin{aligned} &p(\boldsymbol{\theta}|\ \boldsymbol{I},\{\boldsymbol{I}_n\},\boldsymbol{L},\{\boldsymbol{L}_n\}) \\ &\propto p(\boldsymbol{\theta}) \exp\left[\lambda\left(\sum_{\boldsymbol{x}} H\Big[L(\boldsymbol{x})|\ p^d_{\boldsymbol{x}}(L(\boldsymbol{x})|\ \boldsymbol{I};\boldsymbol{\theta})\Big] + \sum_{n=1}^{N} H\Big[L_n(\boldsymbol{x})|\ p^d_{\boldsymbol{x}}(L_n(\boldsymbol{x})|\ \boldsymbol{I}_n;\boldsymbol{\theta})\Big]\right)\right], \end{aligned}$$

where $p(\boldsymbol{\theta})$ is the prior on $\boldsymbol{\theta}$ (e.g., penalty on parameters), H is the cross-entropy function, λ is a constant that weighs the importance of the cross entropy, and $p^d_{\boldsymbol{x}}(l|\ \boldsymbol{I};\boldsymbol{\theta})$ is the soft prediction of the network for label l and image $\boldsymbol{I}$ at location $\boldsymbol{x}$, when the network parameters are equal to $\boldsymbol{\theta}$.

2.2 Inference: proposed method

The goal of the proposed method is to compute the most likely segmentation $\boldsymbol{L}$ of the test image, given the observed variables $\boldsymbol{I},\{\boldsymbol{L}_n\},\{\boldsymbol{I}_n\}$. In a fully Bayesian approach, we would marginalise over the neural network weights $\boldsymbol{\theta}$ when computing the posterior distribution of $\boldsymbol{L}$ that we aim to maximise. However, this

leads to an intractable integral over $\boldsymbol{\theta}$. Instead, we make the standard assumption that the posterior distribution of the parameters $\boldsymbol{\theta}$ is heavily peaked, and therefore we can approximate:

$$\hat{\boldsymbol{L}} = \underset{\boldsymbol{L}}{\operatorname{argmax}}\, p(\boldsymbol{L}|\ \boldsymbol{I}, \{\boldsymbol{I}_n\}, \{\boldsymbol{L}_n\}, \hat{\boldsymbol{\theta}}), \quad \text{with} \quad \hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, p(\boldsymbol{\theta}|\ \boldsymbol{I}, \{\boldsymbol{I}_n\}, \{\boldsymbol{L}_n\}),$$

which we can rewrite as:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{L} p(\boldsymbol{\theta}|\ \boldsymbol{L}, \{\boldsymbol{L}_n\}, \boldsymbol{I}, \{\boldsymbol{I}_n\}) p(\boldsymbol{L}|\ \boldsymbol{I}, \{\boldsymbol{I}_n\}, \{\boldsymbol{L}_n\}) =$$

$$= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{\boldsymbol{x}\in\Omega} \prod_{n'=1}^{N} \Big[p^d_{\boldsymbol{x}}\Big(L_{n'}(\boldsymbol{x})|\ \boldsymbol{I}_{n'}; \boldsymbol{\theta}\Big)\Big]^{\lambda} \sum_{l} \Big[p^d_{\boldsymbol{x}}\Big(l|\boldsymbol{I}; \boldsymbol{\theta}\Big)\Big]^{\lambda} p^f_{\boldsymbol{x}}\Big(l|\ \boldsymbol{I}, \{\boldsymbol{L}_n\}, \{\boldsymbol{I}_n\}\Big) p(\boldsymbol{\theta}).$$

Taking logarithm, we obtain the following objective function:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) = &\log p(\boldsymbol{\theta}) + \lambda \sum_{\boldsymbol{x}\in\Omega} \sum_{n'=1}^{N} \log p^d_{\boldsymbol{x}}\Big(L_{n'}(\boldsymbol{x})|\boldsymbol{I}_{n'}; \boldsymbol{\theta}\Big) + \\ &+ \sum_{\boldsymbol{x}\in\Omega} \log\Big\{ \sum_{l} \Big[p^d_{\boldsymbol{x}}\Big(l|\ \boldsymbol{I}; \boldsymbol{\theta}\Big)\Big]^{\lambda} p^f_{\boldsymbol{x}}\Big(l|\ \boldsymbol{I}, \boldsymbol{L}_n, \boldsymbol{I}_n\Big)\Big\}. \end{aligned} \tag{1}$$

The objective function in Eq. 1 can be optimised with GEM [8]:
E-step. We build a lower bound to the objective function $\mathcal{L}(\boldsymbol{\theta})$ that touches it at the current estimate of the parameters. This involves computing a soft segmentation $w_l(\boldsymbol{x})$ at each pixel of the test image $\boldsymbol{I}$:

$$w_l(\boldsymbol{x}) = \Big[p^d_{\boldsymbol{x}}(l|\ \boldsymbol{I}, \boldsymbol{\theta})\Big]^{\lambda} p^f_{\boldsymbol{x}}(l|\ \boldsymbol{I}, \boldsymbol{L}_n, \boldsymbol{I}_n) \Big/ \sum_{l'} \Big[p^d_{\boldsymbol{x}}(l'|\ \boldsymbol{I}, \boldsymbol{\theta})\Big]^{\lambda} p^f_{\boldsymbol{x}}(l'|\ \boldsymbol{I}, \boldsymbol{L}_n, \boldsymbol{I}_n). \tag{2}$$

M-step. We update the estimates of the network parameters by optimising the bound with respect to $\boldsymbol{\theta}$. Leaving aside terms independent of $\boldsymbol{\theta}$, we seek to maximise:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{\boldsymbol{x}\in\Omega} \sum_{n'=1}^{N} \log p^d_{\boldsymbol{x}}\Big(L_{n'}(\boldsymbol{x})|\ \boldsymbol{I}_{n'}; \boldsymbol{\theta}\Big) + \sum_{\boldsymbol{x}\in\Omega} \sum_{l} w_l(\boldsymbol{x}) \log p^f_{\boldsymbol{x}}(l|\ \boldsymbol{I}, \boldsymbol{\theta}) + \frac{\log p(\boldsymbol{\theta})}{\lambda}. \tag{3}$$

Maximising Eq. 3 amounts to training a neural network with regulariser $\lambda^{-1} \log p(\boldsymbol{\theta})$, using the standard cross entropy loss – and including not only the atlases in the training dataset, but also the target image with its soft segmentation $w_l(\boldsymbol{x})$. This can be achieved with standard numerical techniques, e.g., based on stochastic gradient descent. We note that a standard EM algorithm would require exact maximisation of Eq. 3, whereas numerical methods will only improve the bound. However, improving the bound also leads to an improvement in the original objective function; hence "generalised EM".

The GEM algorithm alternates between the E and M steps until convergence. At that point, it is straightforward to show that:

$$p(\boldsymbol{L}|\ \boldsymbol{I}, \{\boldsymbol{I}_n\}, \{\boldsymbol{L}_n\}, \hat{\boldsymbol{\theta}}) = \prod_{\boldsymbol{x}\in\Omega} w_{L(\boldsymbol{x})}(\boldsymbol{x}),$$

and the final segmentation is given by: $\hat{L}(\boldsymbol{x}) = \operatorname{argmax}_l w_l(\boldsymbol{x})$.

2.3 Model instantiation

In our semi-automated histology segmentation problem, labelled sections play the role of atlases, whereas $\boldsymbol{I}$ is an unlabelled section. To model $p^f_{\boldsymbol{x}}$, we choose the local label fusion model from [9], which relies on a latent discrete field $M(\boldsymbol{x})$ that indexes what atlas generates the test image and its segmentation at each location. The model further assumes that the image intensities $\boldsymbol{I}$ and labels $\boldsymbol{L}$ are conditionally independent given the field $\boldsymbol{M}$. As in [9], we use a Gaussian likelihood term for the image intensities and a LogOdds model based on the signed distance transform for the labels. In addition, we use a prior for the field $\boldsymbol{M}$ that reflects lower reliability of the registration for sections at larger distances from one another, independently from the 2D location $\boldsymbol{x}$: $p(M(\boldsymbol{x}) = n) \propto \exp(-\alpha|z - z_n|)$, where z and z_n be section indices for the test image and atlas n, respectively, and α is a parameter controlling the sharpness of the prior. Following [9], the posterior probability for the labels is then:

$$p^f_{\boldsymbol{x}}(L(\boldsymbol{x})|\ \{\boldsymbol{I}_n\}, \{\boldsymbol{L}_n\}, \boldsymbol{I}) = \frac{\sum_{n=1}^{N} \mathcal{N}[I(\boldsymbol{x}); I_n(\boldsymbol{x}), \sigma^2] e^{\rho D_{\boldsymbol{x}}[L(\boldsymbol{x}); \boldsymbol{L}_n]} e^{-\alpha|z-z_n|}}{\sum_{n=1}^{N} e^{-\alpha|z-z_n|} \mathcal{N}[I(\boldsymbol{x}); I_n(\boldsymbol{x}), \sigma^2] \sum_l e^{\rho D_{\boldsymbol{x}}[l; \boldsymbol{L}_n]}} \quad (4)$$

where $\mathcal{N}$ is the Gaussian distribution; $D_{\boldsymbol{x}}$ is the signed distance transform evaluated at location $\boldsymbol{x}$; and σ^2 and ρ are the likelihood parameters.

For the deep learning framework we use a fully convolutional network (FCN) [10]. We built a FCN on top of a VGG-16 architecture [11], with publicly available weights pre-trained on ImageNet [12]. This architecture was modified by removing the classification layer, and converting fully connected layers to convolutions. A 1×1 convolution layer, with as many channels as output classes, was added at each of the coarse output locations, followed by deconvolution layers to upsample the coarse outputs to fine-grain outputs. Skip connections were added between lower and higher layers, enabling prediction at input resolution. Finally, we used an L2 norm penalty on the network weights, i.e., as $-\log p(\boldsymbol{\theta})$.

3 Experiments and results

3.1 Data

We used the publicly available Allen atlas [2], which includes 106 (unevenly spaced) Nissl-stained histological sections of a human hemisphere with associated manual segmentations for 862 brain structures. The sections are 50 µm thick and digitised at 1 µm resolution, but we downsampled them to 250 µm – as a compromise between detail and computational requirements. Using the label ontology from the Allen Institute, we created two simplified sets of labels: one at the tissue type level (white matter, grey matter, cerebrospinal fluid), and another at the whole structure level, including: cerebral WM, cerebral cortex, lateral ventricle, cerebellar WM, cerebellar cortex, thalamus, caudate, putamen, pallidum, brainstem, hippocampus and amygdala. The tissue level labels are useful for coarse fine-tuning, and the structure level labels will be used in evaluation: using the full Allen ontology introduces excessive noise in the results, due to large differences in the sets of structures appearing in consecutive sections.

Table 1. Minimum and median pixel classification accuracy. The p values are for a non-parametric, paired, two-sided Wilcoxon statistical test comparing medians.

Method	Min. acc. (%)	Med. acc. (%)	p-val vs. CNN	p-val vs. LF	p-val vs. *ad hoc*
CNN	75.35	90.55	N/A	N/A	N/A
Lab. Fus.	73.61	91.81	0.03	N/A	N/A
Ad hoc	85.30	92.62	5×10^{-15}	9×10^{-6}	N/A
Proposed	85.83	92.91	3×10^{-15}	5×10^{-8}	4×10^{-8}

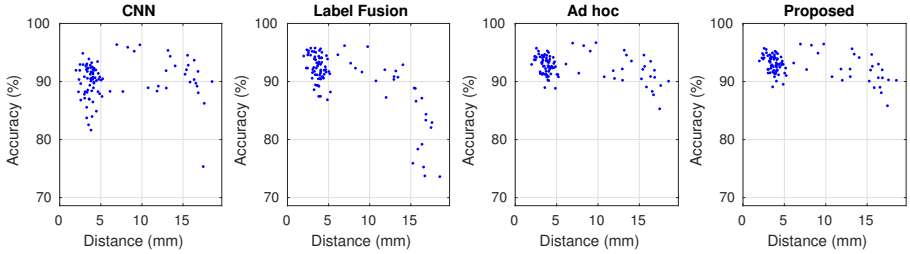

Fig. 2. Pixel classification accuracy vs. average distance between atlas and test sections.

3.2 Experimental setup

We perform a cross-validation on the 106 labelled sections using two folds: one in which even sections are used to predict the segmentation of the odds sections, and vice versa. We compared our proposed method with three competing approaches: the local label fusion method, the CNN alone (with global and local fine-tuning), and an *ad hoc* combination of the two using the product rule, i.e., $p(L) \propto p^d(L)p^f(L)$. As metric of performance, we used the percentage of correctly classified pixels based on the labelling at the structure level.

Each fold was processed as follows. First we globally fine-tuned the network using all available labelled sections of the training fold and the tissue type labels. This enabled a fast transition from the ImageNet weights, effectively adapting the features to the histological images. The learning rates of the final four convolutional layers were increased by a factor of 20 for fine-tuning. We used rotation, translation, scaling and contrast/brighness changes for data augmentation. Then, we visit one unlabelled section at the time, and go over the following three steps: label fusion, local fine-tuning and GEM. For the label fusion, we used the preceding and succeeding labelled sections as atlases. We used NiftyReg with stationary velocity field parameterisation [13] for the registration (default parameters with local correlation metric), and computed soft predictions for the structure level labels with Eq. 4. The local fine-tuning used the same two sections as training data. We replaced the final layers from the globally fine-tuned network, and further fine-tuned to the structure level labels, with the same augmentation scheme. Finally, we iterated between the E and M steps of the GEM algorithm (Eqns. 2 and 3) to produce the final output. The parameters were kept constant for all experiments: $\sigma^2 = 400$, $\rho = 21\,\mathrm{mm}^{-1}$, $\alpha = 1\,\mathrm{mm}^{-1}$, $\lambda = 2$.

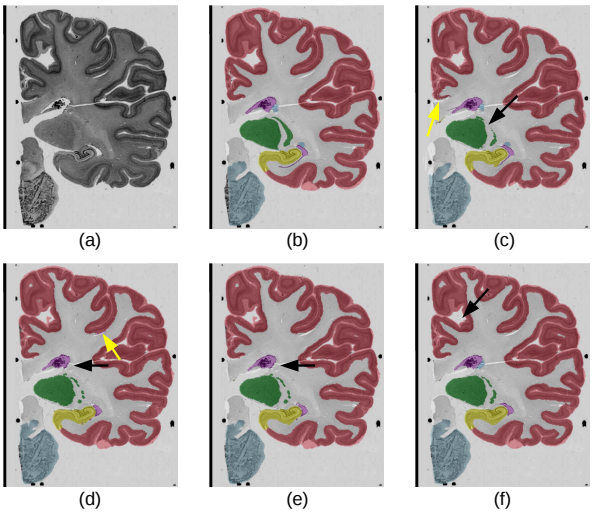

Fig. 3. Sample section from the Allen dataset with segmentation results overlaid. (a) Histological section. (b) Manual segmentation. (c) Label fusion. (d) CNN. (e) *Ad hoc* combination with product rule. (f) Proposed method.

3.3 Results

Table 1 shows the median pixel classification accuracy for the competing methods, computed inside a mask obtained by dilating 2mm the union of the ground truth segmentations. The table also shows p-values for paired, non-parametric tests comparing the medians achieved by the different methods, as well as the minimal accuracy across the dataset – which is a measure of robustness.

Fig. 2 plots accuracy against the mean separation between the atlas and test sections $(1/2)(|z - z_1| + |z - z_2|)$. The CNN provides consistent performance across distances, remaining robust even at large separations. Label fusion alone, in contrast, yields higher scores at low separations (when registration is generally more accurate) but falters at larger distances. Combining the algorithms enables us to take advantage of the strengths of both: the *ad hoc* method outperforms CNN and label fusion, and a further improvement is obtained when integrating the two approaches into a unified model in a principled way. Albeit small (0.3% accuracy), this improvement is consistent ($p < 10^{-7}$) and visually noticeable.

The differences between the methods are illustrated in Fig. 3. Label fusion fails to segment the retrosplenial cortex (Fig. 3c, yellow arrow) and to recover the reticular nucleus of the thalamus (black arrow). The CNN (Fig. 3d) ameliorates these issues, but introduces new errors, e.g., voxels labeled as ventricle due to tears (yellow arrow), or completely missing the caudate due to insufficient contrast (black arrow). The *ad hoc* method (Fig. 3e) solves some of these problems, but still fails to recover the caudate (black arrow). Our approach not only manages to segment the caudate, but also cleans up some other segmentation errors, e.g., the false positives in cortical areas (black arrow in Fig. 3f).

4 Discussion and conclusion

We have presented a probabilistic model for semi-automated segmentation of stacks of 2D histological sections, which allows to incorporate label fusion techniques with deep learning. The model is flexible both in terms of CNN architecture and label fusion methods – as long as the posterior distribution of the segmentation factorises over voxels, which is the case for most available algorithms. Since each iteration requires fine-tuning the network in the M-step (which takes ca. 4 minutes on a Titan Xp GPU), the method is computationally expensive. However, this is seldom a problem in practice because the algorithm can be run offline. Future work will focus on integrating the registration with the segmentation in the framework, such that the registration of atlas sections further away from the test image can benefit from the more robust CNN classification.

Acknowledgement: supported by the EPSRC (CDT in Medical Imaging, EP/L016478/1), ERC (Starting Grant 677697) and NVIDIA (donation of GPU).

References

1. Amunts, K., Lepage, C., Borgeat, L., Mohlberg, H., Dickscheid, T., et al.: BigBrain: an ultrahigh-resolution 3D human brain model. Science **340** (2013) 1472–1475
2. Ding, S.L., Royall, J.J., Sunkin, S.M., Ng, L., Facer, B.A., et al.: Comprehensive cellular-resolution atlas of the adult human brain. Journal of Comparative Neurology **524**(16) (2016) 3127–3481
3. Grady, L.: Random walks for image segmentation. IEEE transactions on pattern analysis and machine intelligence **28**(11) (2006) 1768–1783
4. Criminisi, A., Sharp, T., Blake, A.: GeoS: Geodesic image segmentation. In: European Conference on Computer Vision. (2008) 99–112
5. Rohlfing, T., Brandt, R., Menzel, R., Maurer Jr, C.R.: Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. NeuroImage **21**(4) (2004) 1428–1442
6. Iglesias, J.E., Sabuncu, M.R.: Multi-atlas segmentation of biomedical images: a survey. Medical image analysis **24**(1) (2015) 205–219
7. Çiçek, Ö., Abdulkadir, A., Lienkamp, S., Brox, T., Ronneberger, O.: 3D U-net: learning dense volumetric segmentation from sparse annotation. MICCAI 2016 424
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society. (1977) 1–38
9. Sabuncu, M.R., Yeo, B.T., Van Leemput, K., Fischl, B., Golland, P.: A generative model for image segmentation based on label fusion. IEEE transactions on medical imaging **29**(10) (2010) 1714–1729
10. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. (2015) 3431–3440
11. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. (2009) 248–255
13. Modat, M., Daga, P., Cardoso, M.J., Ourselin, S., Ridgway, G.R., Ashburner, J.: Parametric non-rigid registration using a stationary velocity field. In: Mathematical Methods in Biomedical Image Analysis. (2012) 145–150