Automated three-dimensional detection and counting of neuron somata

Marcel Oberlaender [a],[*],[1], Vincent J. Dercksen [b],[1], Robert Egger [a], Maria Gensel [b],
Bert Sakmann [a], Hans-Christian Hege [b]

[a] Max Planck Institute of Neurobiology, Group "Cortical Column in silico", Am Klopferspitz 18, Martinsried, 82152 Germany
[b] Zuse Institute Berlin, Department of Visualization and Data Analysis, Takustrasse 7, Berlin, 14195 Germany

ARTICLE INFO

ABSTRACT

We present a novel approach for automated detection of neuron somata. A three-step processing pipeline is described on the example of confocal image stacks of NeuN-stained neurons from rat somato-sensory cortex. It results in a set of position landmarks, representing the midpoints of all neuron somata.

In the first step, foreground and background pixels are identified, resulting in a binary image. It is based on local thresholding and compensates for imaging and staining artifacts. Once this pre-processing guarantees a standard image quality, clusters of touching neurons are separated in the second step, using a marker-based watershed approach. A model-based algorithm completes the pipeline. It assumes a dominant neuron population with Gaussian distributed volumes within one microscopic field of view. Remaining larger objects are hence split or treated as a second neuron type.

A variation of the processing pipeline is presented, showing that our method can also be used for co-localization of neurons in multi-channel images. As an example, we process 2-channel stacks of NeuN-stained somata, labeling all neurons, counterstained with GAD67, labeling GABAergic interneurons, using an adapted pre-processing step for the second channel.

The automatically generated landmark sets are compared to manually placed counterparts. A comparison yields that the deviation in landmark position is negligible and that the difference between the numbers of manually and automatically counted neurons is less than 4%. In consequence, this novel approach for neuron counting is a reliable and objective alternative to manual detection.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

The number of neurons in the brain and their varying density between different brain regions is thought to be a fundamental determinant of brain function (Donaldson, 1895; Williams and Herrup, 1988; Williams and Rakic, 1988). During the last 60 years, great effort has been made to estimate neuron densities quantitatively, first addressed by Abercrombie's article "Estimation of Nuclear Population from Microtome Sections" (Abercrombie, 1946). The estimation of absolute numbers of neurons, densities or rates of density change in neuron populations is usually based on random, sparse sampling methods (Cragg, 1967; Rockel et al., 1980) such as stereology (Sterio, 1984). These methods determine cell densities by inspecting a representative sub-volume of tissue and extrapolating the obtained density values to a reference volume. Usually these density values are given with an accuracy of about 10% (Beaulieu, 1993) for large anatomical units, such as primary visual (V1), somato-sensory (S1) or motor cortex (M1). However,

the deviation between densities in previous studies is much larger. For instance in V1 a variety of densities values are reported (40,000 (Cragg, 1967; Knox, 1982; Werner et al., 1982), 52,000 (Beaulieu, 1993), 75,000–80,000 (Peters et al., 1985; Warren and Bedi, 1984) neurons per mm^3). It is hence difficult to determine density changes within or between neuron populations, or across functional subunits, such as a cortical column in S1 (Helmstaedter et al., 2007). In consequence, it would be favorable to count the absolute number of neurons in large volumes ($\sim$0.5 mm^3 for a cortical column in S1 (Meyer et al., in preparation-b)) and hence derive the detailed three dimensional neuron distribution of the brain area of interest.

Recently available three-dimensional imaging techniques (mosaic/optical-sectioning confocal laser scanning or mosaic/optical-sectioning widefield microscopy) and suitable neuronal stains opened new possibilities for the determination of neuronal densities within entire volumes. Neuronal stains, like NeuN (Gittins and Harrison, 2004; Kumar and Buckmaster, 2007; Mullen et al., 1992; Wolf et al., 1996) labeling all neuron somata, or GAD67 (Kaufman et al., 1991; Meyer et al., in preparation-b; Muzio et al., 2002; Staiger et al., 2002) labeling GABAergic interneuron somata, as well as genetically encoded labels of specific neuron populations in transgenic mice or drosophila (Akemann et al., 2004; Luo et al., 2008) allow in principle the quantitative determination of density

---

* Corresponding author. Tel.: +49 89 8578 3285; fax: +49 89 8578 3300.
 E-mail address: oberlaender@neuro.mpg.de (M. Oberlaender).
[1] Equally contributing authors.

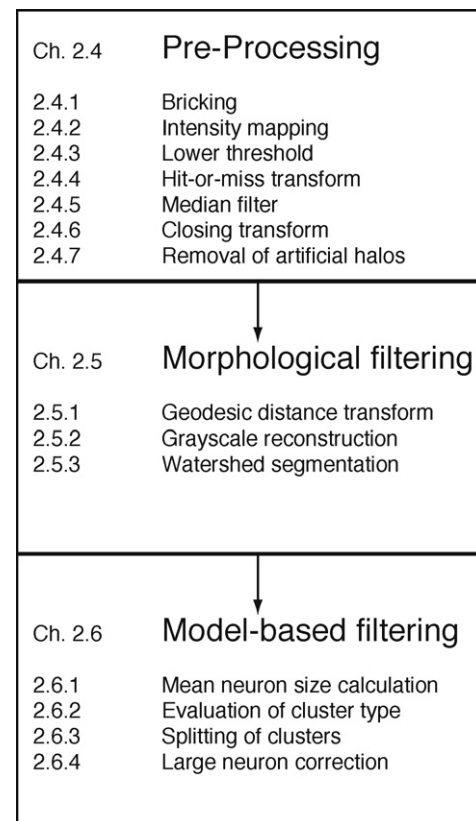differences between neuron populations at high level of detail (e.g. between or within cortical layers).

Several neuron counting and detection methods have been reported, both manual and automated ones. The obvious disadvantage of manual neuron detection, apart from possible subjectivity, is the amount of time needed for neuron counting. In consequence, automated accurate detection and segmentation of neurons from microscopic images has been extensively studied (Liu et al., 2008). In general, these algorithms can be divided into three categories: threshold-based (Wu et al., 2000; Wu et al., 1995), watershed-based (Lin et al., 2003; Lin et al., 2005; Malpica et al., 1997; Nilsson and Heyden, 2005; Vincent and Soille, 1991) and model-based approaches (Chang and Parvin, 2006; Li et al., 2006; Lin et al., 2007; Lin et al., 2005; Raman et al., 2007; Ranzato et al., 2007).

None of the three described automated algorithm categories yield satisfying results for the here presented sample data. This is due to the fact that model-based methods have comparatively better specificity in detecting the targets, i.e. such methods find all objects satisfying the model-shape but only those. In contrast, threshold- or watershed-based approaches display relatively better detection sensitivity, i.e. they find all objects, but usually result in incorrect numbers (Liu et al., 2008), e.g. for our sample data touching, densely clustered neurons are counted as one, resulting in about 20% less neurons.

Here we present an automated 3D neuron counting approach that combines all three approaches to a novel high-throughput system for detection of neuron somata (alternatively called neurons throughout this article). The system is described on the example of confocal image stacks of NeuN-labeled neurons from rat primary somato-sensory cortex (S1). The slightly adapted method for co-localization in multi-channel images is described on the example of NeuN neurons counterstained with GAD67. This allows to measure the fraction of GABAergic interneurons among all neurons (Meyer et al., in preparation-b). We will show that our algorithms are robust and adjustable to different microscope magnifications ($40\times$, $63\times$), and various imaging (confocal, two-photon, widefield) and staining (NeuN, GAD67, Ca^{2+}-sensitive dye) techniques.

The presented processing pipeline consists of three steps and is summarized in Fig. 1. The goal of the first threshold-based step is to create a binary image separating foreground (i.e. stained neurons) from background. It consists of a number of image processing steps, including compensation for imaging or staining artifacts such as bleaching, shading or uneven uptake of the stain, and binarization by local thresholding. This step is data-specific. We will show that the pre-processing algorithm for confocal stacks of NeuN-stained neurons can be systematically adapted to different staining or imaging methods. This is illustrated by two examples: widefield stacks of NeuN-stained neurons and *in vivo* two-photon (2p) stacks of neurons containing Ca^{2+}-sensitive dye. This threshold-based approach is usually not sufficient to detect the true number and position of neurons. High neuron densities and limited microscope resolution result in clusters of neurons that cannot be separated by the local threshold step. The first processing step is therefore regarded as a pre-processing step that guarantees a similar input to the second (watershed-based) and third (model-based) processing steps. The implementation of the latter two steps is independent of the data type.

In the second watershed-based step, clusters of neurons which are connected by narrow links are separated by a morphological filtering process, resulting in an image of distinct watershed regions (3D objects of connected foreground voxels, identified by a label number), and ideally representing individual neurons. Some clustered neurons appear however like a single, large and uniformly stained neuron. The morphological filters are not capable of splitting such clusters into distinct watershed objects.

Ch. 2.4	Pre-Processing
2.4.1	Bricking
2.4.2	Intensity mapping
2.4.3	Lower threshold
2.4.4	Hit-or-miss transform
2.4.5	Median filter
2.4.6	Closing transform
2.4.7	Removal of artificial halos

Ch. 2.5	Morphological filtering
2.5.1	Geodesic distance transform
2.5.2	Grayscale reconstruction
2.5.3	Watershed segmentation

Ch. 2.6	Model-based filtering
2.6.1	Mean neuron size calculation
2.6.2	Evaluation of cluster type
2.6.3	Splitting of clusters
2.6.4	Large neuron correction

Fig. 1. Summary of the image processing pipeline. A threshold-based pre-processing step compensates for imaging and staining artifacts, guaranteeing similar input quality to the second (watershed-based) and third (model-based) processing steps. In contrast to the first step, these steps are independent of the type of image acquisition. They are essential for splitting up densely clustered neurons. By adapting the pre-processing step to other imaging and staining techniques, this pipeline can reliably detect neuron somata from a variety of preparations.

The third, model-based processing step addresses this problem. We assume a single dominant neuron population within the image stacks with a Gaussian-distributed neuron (i.e. soma) volume. The mean neuron volume and its variance are calculated from a volume histogram of the watershed regions. Undivided clusters are then split according to their volume, assuming that it has to be an integer multiple of the mean soma volume. An additional advantage of this constraint is that its parameters are not specified by the user but automatically calculated during the image processing. In a correction step, eventually present spatially separated neuron subpopulations with larger mean volumes are investigated and remain unsplit.

We also present an extension of our method to multi-channel image stacks for co-localization of counterstained neurons. Here NeuN-labeled neurons were counterstained with GAD67 in order to measure the fraction of GABAergic interneurons among all neurons (Meyer et al., in preparation-b).

In the results section, we compare our detection algorithm with manual counts. We show that the automated system reproduces manual counts with more than 90% precision. The 10% deviations originate from detection of ambiguous objects, meaning touching neurons that could for instance be counted as one or two neurons. However, manually counted but automatically missed neurons (false negative, FN) as well as automatically detected but manually missed neurons (false positive, FP) comprise about 5%, respectively. This compensatory effect results in average relative counting differences of approximately 1%. The average absolute deviation in neuron numbers was determined as less than 4%. In addition the deviations in landmark positions proved to be neg-

Table 1
Comparison of manual and automated neuron counts. 14 image stacks from different cortical layers, at different magnification and acquired with different imaging techniques were compared. Differences in numbers (relative and absolute) and landmark positions are of acceptable quality and similar to inter-user variability of manual counts.

| Stack #	Manual #s	Automatic #s	Relative # difference [%]	Absolute # difference [%]	Position deviation [µm]
NeuN confocal 63×					
1	430	425	1.16	1.16	4.54
NeuN confocal 40×					
2	846	896	−5.91	5.91	3.93
3	570	594	−4.21	4.21	4.23
4	871	915	−5.05	5.05	4.74
5	1219	1237	−1.48	1.48	4.75
6	809	776	4.08	4.08	1.92
7	719	670	6.82	6.82	2.05
8	739	680	7.98	7.98	2.07
9	1035	983	5.02	5.02	2.70
10	862	846	1.86	1.86	2.96
11	975	976	−0.10	0.10	3.16
NeuN widefield 40×					
12	311	312	−0.32	0.32	3.81
Mean			0.82	3.67	3.41
Std			4.57	2.63	1.07
Ca^{2+} 2 photon 40×					
13	275	277	0.72	0.72	6.81
14	545	552	1.27	1.27	6.83
Mean			1.00	1.00	6.82

ligible. Further, the automated approach is much faster, reducing manual labor of approximately 4 h per stack to a few minutes and additional computing time of about 1 h.

2. Materials and methods

2.1. Image acquisition

All presented image data was acquired by methods described in detail before. Development and validation of the automated counting pipeline was performed on confocal (Amos et al., 1987) image stacks of NeuN/GAD67-stained neurons (Kaufman et al., 1991; Meyer et al., in preparation-b), kindly provided by Hanno Sebastian Meyer. These stacks were acquired from 50 µm thick physical vibratome sections from cortical tissue of adult wistar rats, cut either along the thalamocortical, coronal or tangential axis. Large three-dimensional confocal image stacks were generated by mosaic/optical-sectioning (Agard, 1984; Oberlaender et al., 2007) using a confocal laser scanning microscope (Leica SP2, Leica Microsytems GmbH, Wetzlar, Germany). Images were acquired
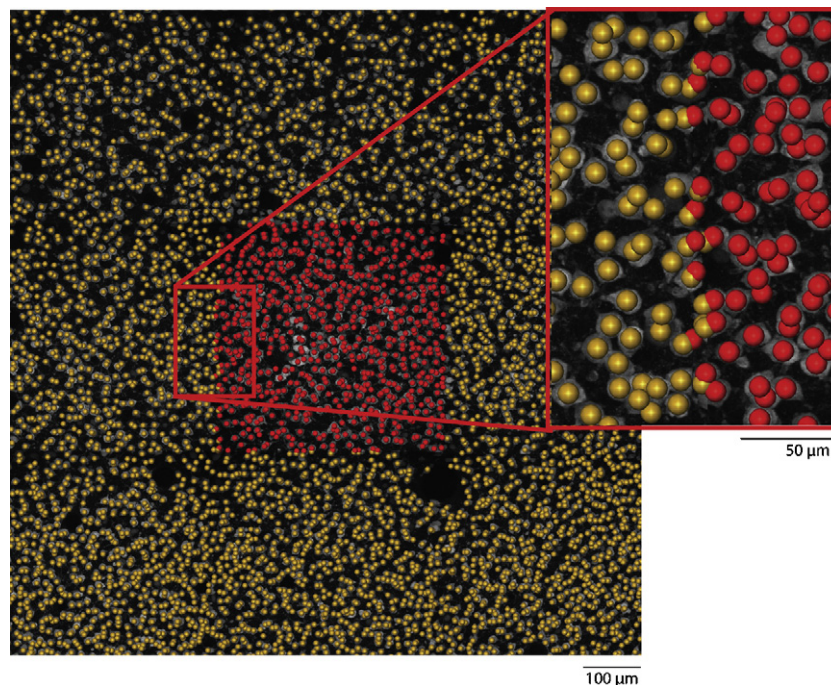
Fig. 2. Treatment of *x/y* border effects. Neurons at the *x/y* border of image stacks that are part of large mosaic images (red landmarks represent one field of view) are always counted, independent whether more than 50% of the neuron soma volume is within the stack. The overlap between adjacent mosaic tiles is approximately 5 µm. Coinciding neuron landmarks in the overlap area are erased (red/yellow). By choosing the mosaic area larger than the region of interest, *x/y* border effects can be neglected completely.

10 μm x 10 μm x 10 μm
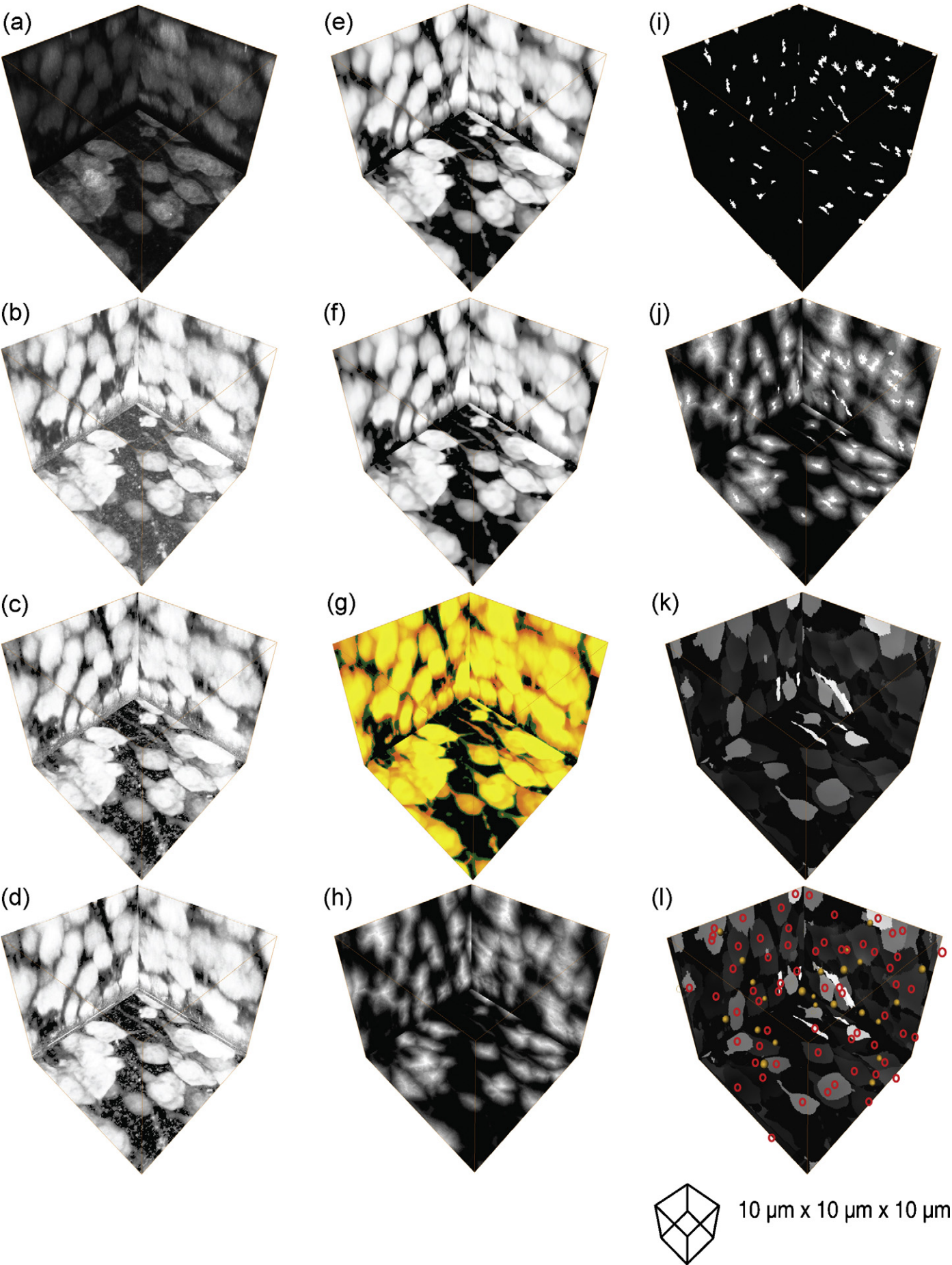
by 40× (HCX PL APO CS; N.A.: 1.25–0.75; oil-immersion) or 63× (HCX PL APO CS; N.A.: 1.3; glycerol-immersion) magnification objectives, yielding a pixel size of 0.366 μm × 0.366 μm for the subsequently presented stacks 2–11 (Table 1) or 0.232 μm × 0.232 μm (stack 1, Table 1), respectively. Mosaic refers to multiple overlapping 1024 × 1024 pixel images (e.g. 4 × 4), each representing one microscopic field of view. Sampling along the z-direction during optical-sectioning was 0.61 μm. The subsequent pipeline was always applied to single fields of view (i.e. 1024 × 1024 pixel wide image stacks).

Two-photon (2p) image stacks of neuron somata containing Ca^{2+}-sensitive dye (Gobel et al., 2007) were acquired by an 40× objective (Olympus, LUM PLAN FL/IR; N.A.: 0.8), pre-processed (subtraction of fluorescent channel containing astrocytes; personal communication with Dr. Fritjof Helmchen) and kindly provided by Dr. Fritjof Helmchen and Dr. Werner Goebel. Depending on the camera settings, the resulting voxel dimensions were 0.776 μm × 0.776 μm × 5 μm and 0.61 μm × 0.61 μm × 2.5 μm for stacks 13 and 14 (Table 1), respectively.

Widefield image stacks of NeuN-stained cortical slices, provided by Dr. Andreas Frick and Dr. Melanie Ginger were acquired by mosaic/optical-sectioning microscopy (Olympus BX-51, Olympus, Japan) using a 40× objective (U PLAN FL N, N.A.: 1.3; oil-immersion), which has been described in detail before (Oberlaender et al., 2007). The according voxel size was 0.232 μm × 0.232 μm × 0.5 μm for stack 12 (Table 1).

2.2. Manual detection of soma positions

Image stacks containing single mosaic tiles were loaded into AMIRA 4.0 or 4.1 (Mercury-Computer-Systems, 2007; Stalling et al., 2005). Landmarks (3D voxel coordinates) were assigned manually to the center of all stained neuron somata (soma landmarks) during a careful examination of the image planes (optical sections).

Objects at the stack border in the x- or y-direction were always counted. The x/y overlap was set to approximately 5 μm. In consequence neurons at the x/y border (overlap area) of mosaic tiles were detected twice. By aligning the mosaic images, twice detected neuron somata coincide and can be erased (Fig. 2). If the mosaic area is chosen larger than the area of interest, x/y border effects can hence be completely neglected.

The x/y border rule is not applied at the z-borders of the image stacks. Here neurons are regarded to be within the image stack if their diameter increases to a maximum value and decreased again or was constant for three more optical sections before reaching the stack border. A detailed description of the manual counting, the validation of the border criteria and the approximate inter-user-variability of 2.1% can be found in Meyer et al. (in preparation-b).

2.3. Computing hard- and software

The software and algorithms are custom written in C++ (Stroustrup, 2004). The raster image file I/O, iteration through a raster image, as well as multiple of the subsequent image filters use the ITK Image Processing Library (Ibanez et al., 2005). The software is executed on AMD dual-core 64-bit Opteron computers, equipped with either 1 CPU and 2 GB of memory, 4 CPUs and 32 GB of memory (DELTA Computer Products GmbH, Reinbek, Germany)

or 8 CPUs and 64 GB of memory (fms-computer.com, Netphen, Germany).

2.4. Threshold-based filtering (pre-processing)

Fluorescent images can suffer from two kinds of artifacts. First, shading or bleaching of the stain leads to an uneven illumination across the images and is usually caused by the image acquisition itself. It results in different signal-to-noise ratios (SNRs) across and between individual image planes. The second artifact is caused by uneven uptake of the fluorescent dye, resulting in varying intensity values across individual neurons. The first issue is addressed by subdividing each image plane into rectangular bricks and processing each image plane individually as will be described in Section 2.4.2/Section 2.4.3. The second issue is addressed by processing each three-dimensional object (e.g. neuron soma) individually as will be described in Section 2.4.7.

It should be emphasized that the mentioned artifacts are strongly dependent on the imaging instrumentation and the used staining method. Therefore we present examples of how to adapt the pre-processing step to different techniques, namely *in vivo* 2p image stacks of neurons stained with Ca^{2+}-sensitive dyes and NeuN widefield images (in Section 2.4.8). If the imaging system incorporates automatic shading correction that reduces deviations in SNR already during the image acquisition, the subsequent filters will not affect the image quality negatively. However, the here presented imaging systems yield no sufficient intrinsic correction and hence the pre-processing is essential for an authentic neuron counting.

2.4.1. Bricking

Uneven illumination across individual image planes prohibits the application of global image operators. Further, illumination deviations between individual image planes limit a general application of 3D image operators. Each image plane (i.e. one field of view; 1024 × 1024 pixels) is hence subdivided into 2D bricks. In the case of the confocal microscope at 40× magnification the bricks size is 256 × 256 pixels. This size was determined by systematic testing and is linearly adjusted for different magnifications (or pixel resolutions), in order to guarantee a similar brick area (in μm²) for all kinds of images. Hence, the number of bricks decreases with decreasing magnification (or pixel resolution) and fixed image size.
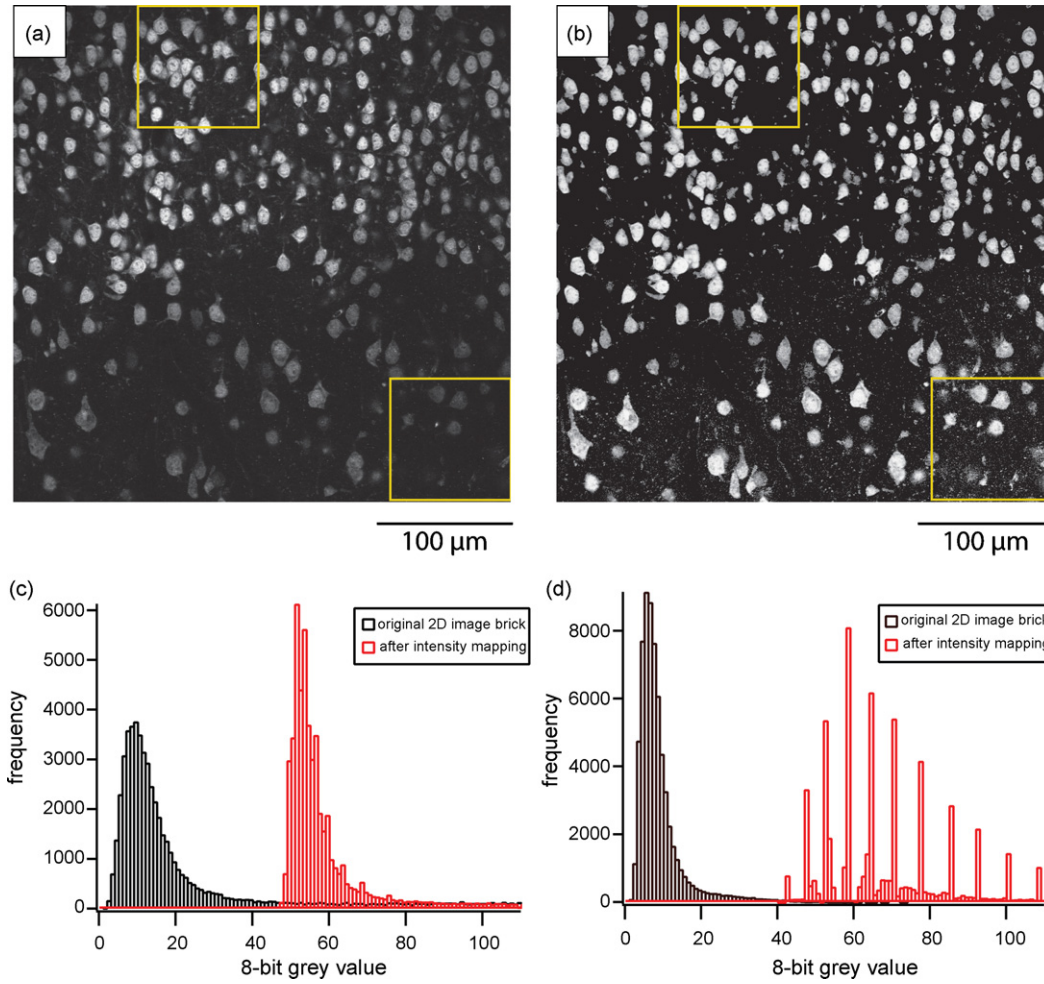
2.4.2. Intensity mapping

Direct application of a lower threshold operator on each 2D brick, which sets all pixels having a value smaller than the threshold value to zero, proved to be problematic. The SNR is usually too low to set an adequate threshold value that separates somata from background. Hence the intensities for each brick are mapped by a non-linear sigmoid-shaped filter (Eq. (1)) (Ibanez et al., 2005) onto a new range:

$$I' = \frac{255}{1 + e^{-(I - \beta/\alpha)}}, \tag{1}$$

where I' and I denote the new and old intensity (gray) values, respectively. β ideally represents the center and α the width of the neuron's intensity range. This filter progressively attenuates intensity values outside this range and produces a very smooth and continuous transition to the specific intensity range of inter-

Fig. 3. Each panel shows x/y-, x/z- and y/z-projections of a representative sub-volume (Fig. 9) of NeuN-stained somata, imaged with a confocal microscope and a 40× objective. Each panel refers to the subsequent processing steps summarized in Fig. 2. (a) Original stack, (b)–(g) pre-processing, (b) local intensity mapping → amplification of weak somata, (c) local lower threshold → reduction of background, (d) hit-or-miss transform → reduction of small isolated noise artifacts, (e) median filter → reduction of noise and smoothing of structures, (f) closing filter → filling of small holes within the structure, (g) cropping filter → structures are cropped from their artificial halos (green), (h)–(k) watershed-based processing, (h) Euclidean distance transform, (i) regional maxima = markers, (j) grayscale reconstruction → starting basins for watershed segmentation, (k) marker-driven watershed segmentation → individual foreground regions representing individual or clustered somata, (l) model-based processing, (l) splitting of remaining clustered watershed objects according to object volume (Fig. 8) → set of position landmarks (orange); for illustration: landmark projections (red circles).

Fig. 4. Illustration of local intensity mapping and local threshold function. Shading and bleaching can cause significant SNR gradients within individual image planes. (a) Single thalamocortical x/y image plane affected by uneven illumination. The high-contrast in the upper brick results in a broad gray-value histogram (panel c; width of black histogram = 6.22), whereas the low-contrast bottom brick has a narrower histogram (panel d; width of black histogram = 3.58). (b) Image plane after local intensity mapping. The neurons across the entire plane are of similar intensities. The noise in the prior low-contrast brick is high, resulting in a much broader histogram (panel d, width of red histogram = 56.52) whereas the width of histogram of the high-contrast brick remains more or less unchanged (panel c, width of red histogram = 4.36). (c) The uneven illumination results in different histogram width after intensity mapping. In bricks with high contrast, the intensity mapping leads to amplification of the somata and attenuation of the background. Hence the lower threshold should be approximately the mean value of the mapped image. (d) In low-contrast images, the structures and a significant part of the background are amplified, resulting in a broad histogram. Hence the lower threshold needs to be higher than the mean value after mapping, which is lowered by amplified background. This is realized by the following threshold function: $t_1 = \mu_{new} + (1.1/\sigma_{original}) \cdot \sigma_{new}$. For high-contrast images t_1 is essentially μ_{new} (here: $\mu_{new} + 1$), whereas in the low-contrast case t_1 is significantly increased (here: $\mu_{new} + 17$).

est (Fig. 3b). It results in a per-brick amplification of the neurons with respect to their surroundings (Fig. 4a/b). Systematic testing yielded an intensity range of neurons that is best described by the following values for α and β:

$$\beta = \mu_{original} + 0.75\sigma_{original}; \qquad \alpha = \sigma_{original}, \qquad (2)$$

where $\mu_{original}$ and $\sigma_{original}$ refer to the mean gray value and standard deviation of each 2D image brick, respectively.

2.4.3. Lower threshold

Once the neurons are amplified with respect to their surrounding, a lower threshold t_1 is applied in order to separate the neuron somata from background:

$$t_1 = \mu_{new} + \frac{1.1}{\sigma_{original}} \cdot \sigma_{new}, \qquad (3)$$

where μ_{new} and σ_{new} refer to the mean gray value and standard deviation of each 2D image brick after intensity mapping. This thresholding step sets all voxels below t_1 to background, i.e. to value zero. The local threshold function comprises a term inversely pro-

portional to the standard deviation of the image brick prior the intensity mapping. This accounts for possible uneven illumination within each image plane. The width of the intensity distribution ($\sigma_{original}$) can vary significantly from one brick to the next. Two extreme situations occur when neurons are surrounded by low background, resulting in broad intensity distributions (Fig. 4a, top brick/Fig. 4c), and when neurons are surrounded by background values similar to the neurons' intensities, resulting in a narrow distribution (Fig. 4a, bottom brick/Fig. 4d). In the latter case, the width of the intensity distribution after mapping will be much larger than the original one ($\sigma_{new} \gg \sigma_{original}$), resulting in an approximate threshold of $t_1 \approx \mu_{new} + \sigma_{new}$ (Fig. 4d). In the other case, the two widths will be of similar order of magnitude, resulting in a threshold of $t_1 \approx \mu_{new}$ (Fig. 4c). Thus, this filter is capable of discriminating neurons surrounded by high and low background values (Figs. 3c and 4b).

2.4.4. Hit-or-miss transform

The two previous filtering steps result in image stacks of significantly reduced background. However, small speckle artifacts

are usually still present. The image planes are therefore subjected to a hit-or-miss transformation with rectangular frame masks of increasing size as structuring elements (Gonzalez and Woods, 2002; Oberlaender et al., 2007). The transformation is applied to every image plane. Isolated foreground objects that are completely surrounded by a frame are converted to background (Fig. 3d). Beginning with a radius of one pixel and increasing the frame size subsequently to three pixels, small and isolated artifacts are removed.

2.4.5. Median filtering

In order to smooth the intensity distribution within neurons, a median filter is applied as implemented by the ITK (Ibanez et al., 2005). Each voxel is assigned a new intensity value which is the median value of its surrounding voxels within a $5 \times 5 \times 5$ voxel large neighborhood (Fig. 3e). The neighborhood size was obtained by systematic testing. This filter is computed in three dimensions because neurons are 3D objects, which consist of 2D planes that may vary systematically in gray values. 2D median filters would not decrease these inter-plane deviations.

2.4.6. Closing transform

Next, a grayscale closing filter (Gonzalez and Woods, 2002) is applied as implemented by the ITK (Ibanez et al., 2005). Its geometrical interpretation is that a "sphere" rolls along the outside boundary of a foreground object (i.e. neuron soma). It tends to smooth contours, fuses narrow breaks, eliminates small holes, and fills small gaps in the neurons (Fig. 3f). The 3D structuring element (sphere) has a radius of five voxels (approximately 2/3 of a soma radius) and was derived after systematic testing.

2.4.7. Removal of artificial halos

The uneven uptake of stain results in neurons of weak intensities in intermediate neighborhood to neurons with high intensity values. The intensity mapping described above causes amplification of the surrounding of such weakly pronounced neurons and sometimes fuses them with other neurons (halos (Fig. 3g, green)). By removing "halos" that were introduced by the intensity mapping, the neurons are cropped to their original volume (Fig. 3g, yellow/orange). This is realized by processing each 3D object of connected foreground pixels individually. For each 2D plane of each
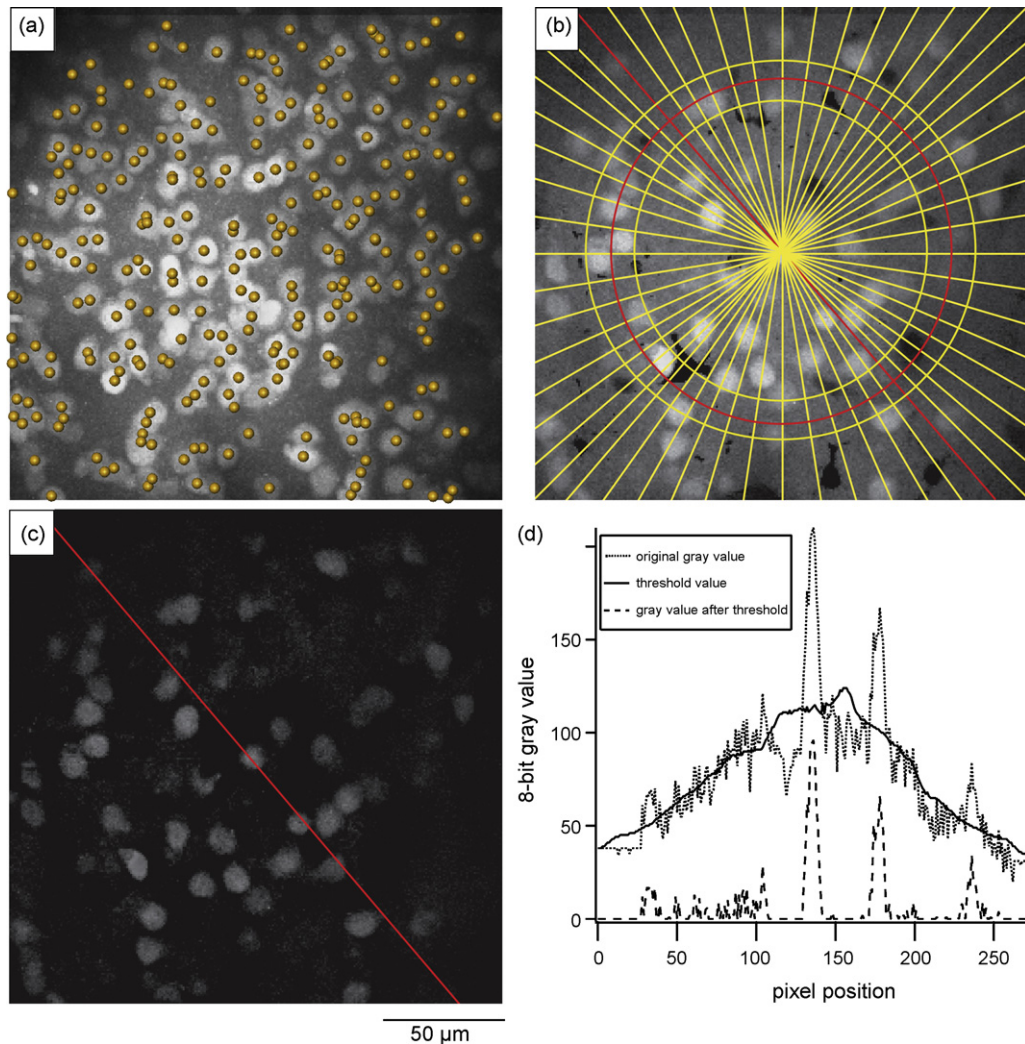
Fig. 5. Illustration of radial threshold of 2p images. Radial symmetric shading results in high-contrast somata at the center and low-contrast somata at the edges. In order to apply the brick-wise local intensity mapping, these shading artifacts need to be minimized. (a) Maximum *z*-projection of 2p image stack of neurons stained with Ca²⁺-sensitive dye superimposed with automatically generated landmarks. (b) 360 Radial lines superimposed on individual image plane from the stack in (a). A lower threshold for each pixel is calculated as the mean value of its 25 neighboring pixels along both line directions (i.e. pixels on the red circle are assigned a threshold value averaged from the pixels on the same line and between the yellow circles). (c) Image plane from (b) after application of the radial threshold function. (d) Intensity profile across the lines in (b) and (c). The final dashed line is the result of subtracting the threshold (solid line) from the original profile (dotted line). After this step the pre-processing can be applied with modified parameters as in the confocal case (Fig. 3).

object, a lower threshold value t_2 is calculated, defining the foreground for this object plane:

$$t_2 = \mu_{\text{filter}} - 1.2 \cdot \sigma_{\text{filter}} + \frac{40}{\sigma_{\text{original}}}, \qquad (4)$$

where μ_{filter} and σ_{filter} refer to the mean gray value and standard deviation of each 2D object plane after the closing filter. The local threshold function comprises a term inversely proportional to the intensity deviation of the 2D object plane σ_{original} in the unprocessed image, again compensating for uneven illumination between the bricks as described in Section 2.4.3. The parameters for the threshold function are obtained by systematic testing. This is the final result of the first (pre-processing) step.

2.4.8. Adaptation of pre-processing to different imaging techniques at various magnifications

The preceding pre-processing pipeline is strongly dependent on the imaging and staining procedure. The only input parameters are the $x/y/z$-resolution of the stacks. In this section we give examples of how to adjust the filters to in vivo 2p image stacks of neurons stained with Ca^{2+}-sensitive dyes and NeuN-stained neurons imaged with a fluorescent widefield microscope.

In contrast to the above shown confocal images, the presented in vivo 2p images usually have high background values in the image center, decreasing in a radial symmetric manner (Fig. 5a). This is compensated for by applying a lower local threshold which is calculated along 360 radial lines for each 2D image plane (Fig. 5b). A lower threshold is computed for each pixel, using the line through the pixel and the image center. The threshold value is defined as the mean intensity of 25 pixels along this line on both sides of the current pixel. This results in image planes with significantly reduced radial symmetric illumination (Fig. 5c).

After this additional filtering step the above described pipeline is applied with different parameters derived after systematic testing ($\beta = 2.2 \times \mu_{\text{original}}$; $\alpha = 0.5\sigma_{\text{original}}$; $t_1 = 100/\sigma_{\text{original}}$; $t_2 = \mu_{\text{filter}} - 1.2 \times \sigma_{\text{filter}} + 100/\sigma_{\text{filter}}$). Using these adjustments the subsequently described processing steps (Section 2.5/Section 2.6), can be applied as in the confocal case.

Due to out-of-focus light, the presented widefield images have significantly lower resolution compared to the confocal stacks shown above (Fig. 6a). Especially along the optical (z-)axis this blurred contribution makes a valid detection and splitting of neuron somata difficult. The out-of-focus light can be significantly reduced (Fig. 6b) by applying a maximum likelihood (MLE) deconvolution filter with a reasonably determined point-spread-function (PSF). The PSF of the used imaging system in combination with cortical tissue has been determined previously (Oberlaender et al., 2009) and is applied by Huygens Professional software (Scientific-Volume-Imaging, 1995–2007). After the deconvolution, the pre-processing pipeline is applied with the same parameters as described in the confocal/NeuN case.

2.5. Morphological filtering

The filtering described in the previous chapter results in a stack of 2D gray-value images, which is then transformed into a binary stack by setting all pixels with a gray value different from 0 to 255. In the following such stacks will be considered as single 3D binary images. In order to find the total number of neurons in the image, one could simply count the total number of 3D connected foreground objects (groups of connected voxels) in the image. However, the limited resolution of light microscopy imaging systems in addition to high neuron densities results in clusters of neurons that cannot be separated by the pre-processing pipeline. In consequence, direct counting produces total neuron numbers which are generally too low, because a single foreground object may consist of
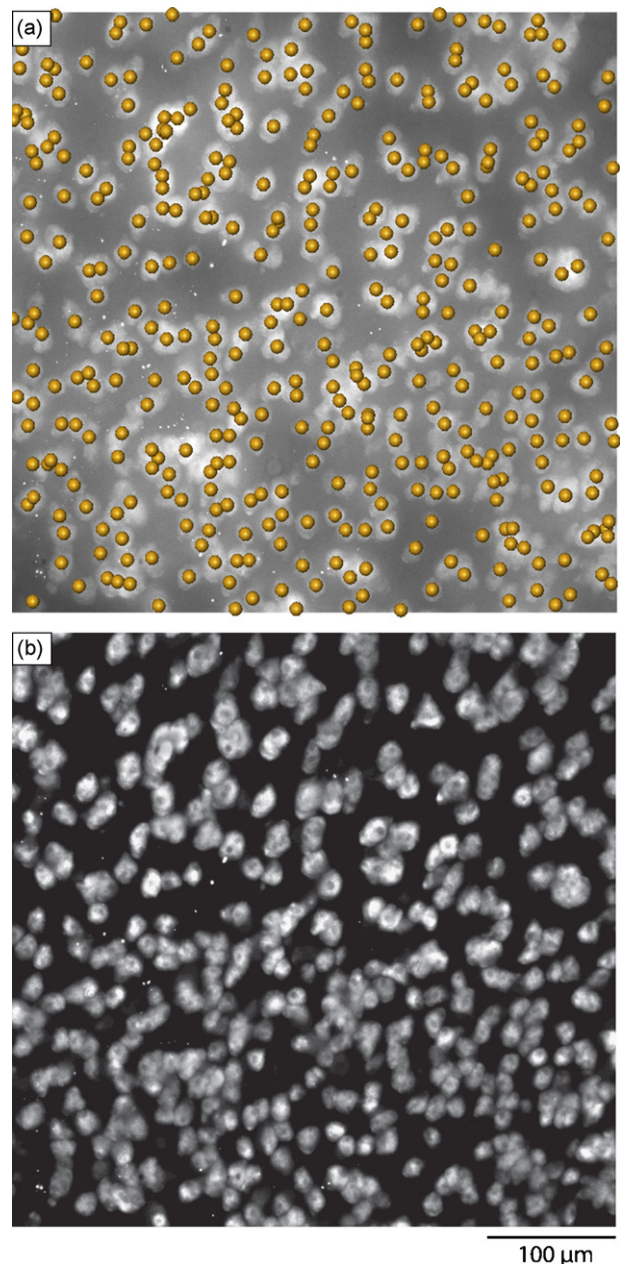

Fig. 6. Adjustment of pre-processing for NeuN-stained neurons imaged with a fluorescent widefield microscope. (a) Maximum z-projection of image stack with superimposed automatically generated landmarks. Out-of-focus light results in a blurred contribution to each image plane, making an automated neuron counting problematic. (b) By application of a MLE deconvolution filter prior to the standard pre-processing pipeline, the out-of-focus light is significantly reduced and the automated detection pipeline can be executed.

multiple connected neurons. We therefore divide such clusters into their constituent neurons, using a method described by Vincent and Dougherty (1994). This method consists of three steps:

1. Computation of a distance transform for each 3D foreground object of the binary image.
2. Finding exactly one marker for each neuron (i.e. multiple markers for neuron clusters), where marker refers to a single voxel or a group of connected voxels.
3. Computation of a watershed transformation, using the markers as initial basins.
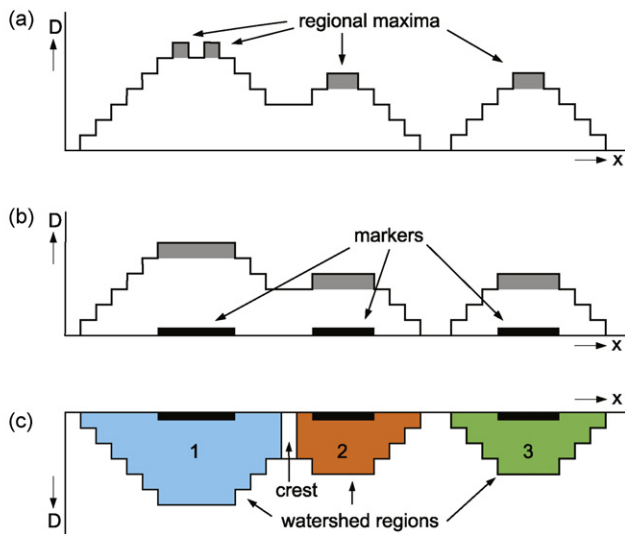
Fig. 7. One-dimensional illustration of the separation of touching somata by marker-based watershed segmentation. Objects containing two (or more) maxima in the distance field separated by a minimum are assumed to consist of two (or more) clustered neurons and need to be separated. (a) Regional maxima of the distance field D. Due to contour irregularities multiple maxima per object may appear (left object), which is undesired. (b) By subtracting 1 from the distance field at the positions of the regional maxima, and computing the regional maxima of this modified distance field, better markers are obtained. (c) The adapted distance field is "flooded" using the marker positions as initial basins. Positions where the levels from different watersheds meet, are marked as crest regions. After the flooding is completed, crest regions are turned into background. (D has been inverted to better illustrate the basin-flooding metaphor.) As a result clustered neurons are split at the minima of the distance field, i.e. where the connection between two neurons is thinnest.

2.5.1. Geodesic distance transform

First, a distance transform (Gonzalez and Woods, 2002) is computed for each 3D foreground object in the binary image. This results in intensity values for each voxel that resemble the physical Euclidean distance to the closest background voxel. Thus, voxels in the interior of objects have high values, object voxels close to the boundary have low values and background voxels have zero value (Fig. 3h). These values are computed by repeatedly (binary) eroding (Gonzalez and Woods, 2002; Ibanez et al., 2005) the objects in the pre-processed image, successively peeling their outer boundaries. During each erosion step and for each foreground object, the physical Euclidean distance to the prior erosion level is assigned to voxels of the current most outer layer. A $3 \times 3 \times 3$ voxel binary cross is used as structuring element for erosion.

2.5.2. Grayscale reconstruction

The second step will ideally generate a single marker for each neuron within a foreground object (Fig. 3i). One approach to realize this would be to compute the regional maxima of the distance image (Fig. 7a). A regional maximum M of a grayscale image I is defined as a connected region of voxels with a given value h (plateau at altitude h), such that every voxel in the neighborhood of M has a value strictly lower than h. An efficient method to compute regional maxima uses a morphological operation called *grayscale reconstruction* (Vincent, 1993) as implemented by the ITK (Ibanez et al., 2005). Computing all regional maxima results however in too many markers: some objects have multiple very close regional maxima, due to contour irregularities and discretization artifacts. Usually these markers differ by only one level in the distance function. We therefore reconnect such regional maxima by subtracting 1 from the values in the distance image at all marker positions and compute the regional maxima (Fig. 3i) of this modified distance function (Fig. 7b).

2.5.3. Watershed segmentation

The final step uses the set of markers M to assign the set of foreground voxels belonging to each neuron. These neuron regions are found using a procedure called *marker-driven watershed segmentation*. The general watershed algorithm uses intensity information to divide a gray-value image into foreground regions (catchment basins), separated by watershed lines (background). This algorithm can be illustrated by a landscape flooding metaphor. The inverted distance image $D' = -D$ can be regarded as a landscape where the minima of D' correspond to valleys (Fig. 7c). When this landscape is flooded, the water level starts to rise from the valleys (or catchment basins) until the different basins meet at the watershed lines (or crests). Afterwards, each basin corresponds to one of the desired neuron regions and is bounded by the watershed lines and/or the image background. The marker-driven watershed segmentation ensures that we obtain exactly one region for each marker, by creating an input image D^* adapted from D' such that (a) its only regional minima are located at the marker positions and (b) its only crest-lines are the highest crest-lines of D' that are located between the minima (marker positions). The image D^* is then flooded as in the general watershed case. For details we refer to Vincent and Dougherty (1994). We use the flooding algorithm described in Soille (1999) in order to efficiently find the neurons corresponding to the markers (Fig. 3k).

2.6. Model-based cluster splitting

The preceding processing steps result in individual 3D objects (neuron regions) that represent the neuronal somata within the image stack. However, some clusters of neurons are still not separated because they have similar intensities and are so close to each other in the original image that they appear like a single, almost spherical neuron. To separate such clusters we chose a model-based filter. We assume that within each stack there is a single dominant neuron population of neurons with Gaussian distributed volumes and that most clusters could be split by the filters of the first two pipeline steps.

### 2.6.1. Volume histogram and mean neuron size calculation

Regarding the assumption above, we calculate the voxel volume of each watershed region and create a histogram of these volumes. The first histogram bin always comprises many small artificial objects. Hence, taking the second peak of the histogram as the mean volume of the dominant neuron population, a Gaussian distribution is fitted to the histogram at this value (Fig. 8a). The distribution for individual neurons (i.e. not part of a neuron cluster) is parameterized by three values, the mean value, the standard deviation and the peak amplitude (μ_1, σ_1, A_1). Clusters of $n = 2, 3, \ldots, N$ connected neurons will also be parameterized by Gaussian distributions. Their according mean value μ_n is simply given by:

$$\mu_n = 1.1 \cdot n \cdot \mu_1. \tag{5}$$

The additional multiplication with 1.1 is due to the assumption that the unresolved gaps between the neurons add a small amount to the cluster volume and is derived by systematic testing. The standard deviation for each cluster type σ_n is calculated by standard error propagation (see standard physics textbooks):

$$\sigma_n = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \sigma_i, \tag{6}$$

and the according amplitude A_n is calculated by averaging the height of the five bins around μ_n. This is done until the largest 3D region (object) in the watershed image is covered by a distribution (Fig. 8a).
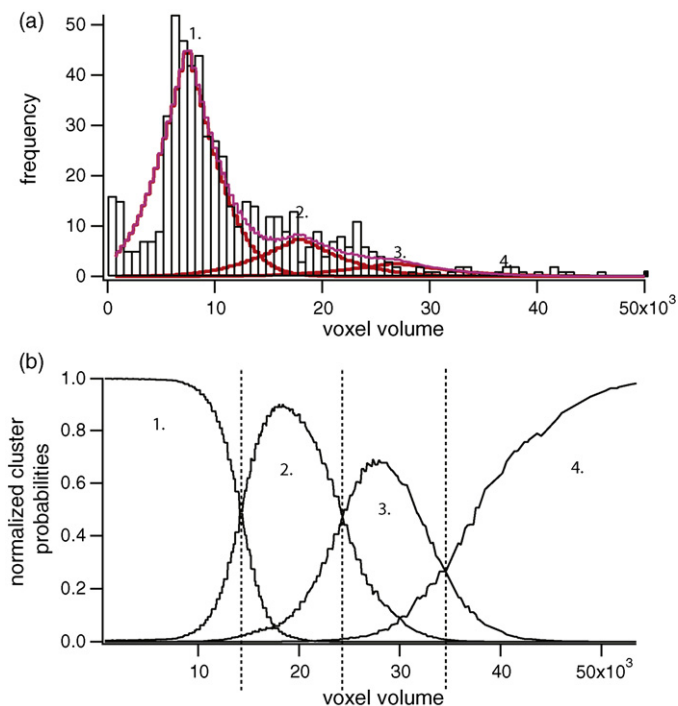
Fig. 8. (a) Volume histogram of watershed objects. The first mode is assumed to reflect the dominating neuron population and is fitted by a Gaussian (red 1). The higher modes are assumed to be integer multiples of the first mode (→ here: 3 more Gaussians representing clusters of 2, 3 and 4 neurons respectively). The pink plot represents the sum of the four distributions. (b) Normalized probability values for each object volume. According to the above distributions four probability values are calculated for each object volume and normalized to 1. The resulting 4 plots are shown. The intersections (vertical dashed lines) represent volumes that separate the clusters.

2.6.2. Evaluation of cluster type

Each watershed object can only belong to a single distribution. Therefore the probability p_{mn} for each object and for each distribution is calculated, where m refers to the object number and n to the nth cluster distribution (number of connected neurons). Each p_{mn} value is normalized to one (Fig. 8b), resulting in N normalized cluster probabilities for each watershed object.

$$\bar{p}_{mn} = \frac{p_{mn}}{\sum_{n=1}^{N} p_{mn}} \qquad (7)$$

Each object is regarded to consist of as many neurons as is its highest normalized probability value. However, objects that are smaller than the mean neuron volume minus twice the standard deviation are considered to be artifacts and ignored during the further processing.

2.6.3. Splitting of clusters by k-means cluster analysis

Once each watershed object is assigned a most likely cluster type (i.e. $n = k = 1, 2, \ldots, N$ neurons), the according k reference voxels (landmarks) are calculated. We therefore use a k-means clustering algorithm as implemented by the ITK (Ibanez et al., 2005). The k-means algorithm works as follows:

1. The input for each watershed object are k (=n; i.e. most likely number of neurons in a cluster) initial mean values (default landmarks) specified as k random voxels within the object.
2. Each voxel of a watershed object is assigned to its closest landmark among the k mean values.
3. Calculation of each k-means cluster's mean from the newly assigned landmark voxels within the objects and hence updating the k mean values of a neuron cluster.

4. Repetition of step 2 and step 3 until the termination criteria is met, here if no voxel changes its cluster membership from the previous iteration.

This results in k landmark voxels for an object (consisting of $k = n$ neurons). Each landmark is used as position reference of an individual neuron and a list of these landmark voxels is visualized in Amira (Mercury-Computer-Systems, 2007; Stalling et al., 2005) (Fig. 3l).

2.6.4. Correction for 2nd population of larger neurons

The assumption of a single Gaussian distributed neuron volume across the entire image stack can cause miss-counting if two neuron populations of significantly different volumes are present. Here we assume that there is only a small spatial overlap between these populations and that one of them is dominant. We argue that this is a reasonable assumption for the presented image sizes of 375 μm × 375 μm × 50 μm (confocal microscope with 40× objective). If the minor population is smaller in volume than the majorities' mean volume (i.e. GABAergic interneurons), these neurons will still be counted as one. However, if the minority population consists of large neurons (Fig. 9) the prior described splitting will result in an overestimation of the neuron density.

This issue is addressed by evaluating the local surrounding of each watershed object before model-based splitting. If more than 10% of the watershed objects in a surrounding box of $300 \times 300 \times 100$ voxels have the same cluster probability (larger than 1), the object will not be split (Fig. 9). This filter also corrects for systematic errors of the pre-processing step. In bricks with low SNR the neurons are slightly enlarged compared to bricks of high SNR. If significant SNR gradients are present within the image stack, the systematic increase in volume in low SNR bricks can lead to an artificial second neuron type. This means that the neuron volume in some regions of the image can be systematically larger compared
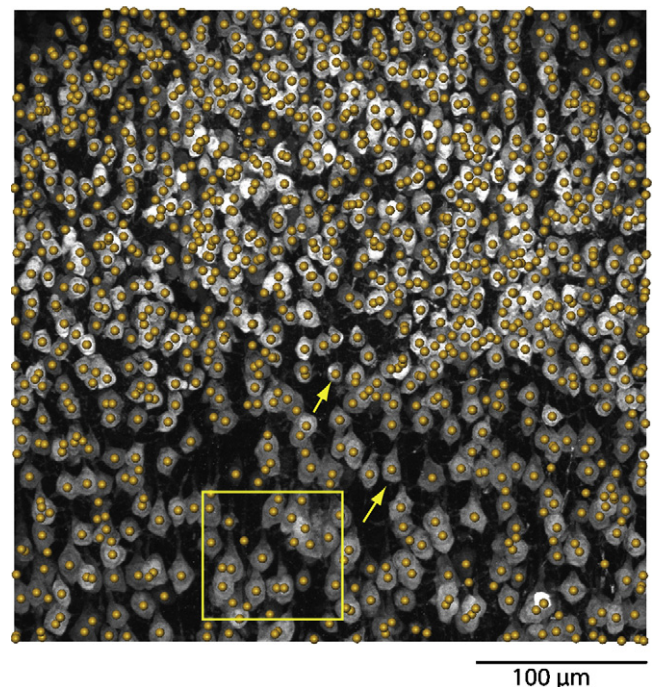
Fig. 9. Maximum z-projection of NeuN-stained confocal image stack. The yellow box represents the representative volume shown in Fig. 3. This stack is in an area of the cortex where layer 4 neurons (upper part) overlap with larger layer 5 neurons (lower part). Even though two neuron populations with significantly different volumes are present, the correction algorithm avoids miscounting (e.g. neurons indicated by arrows).
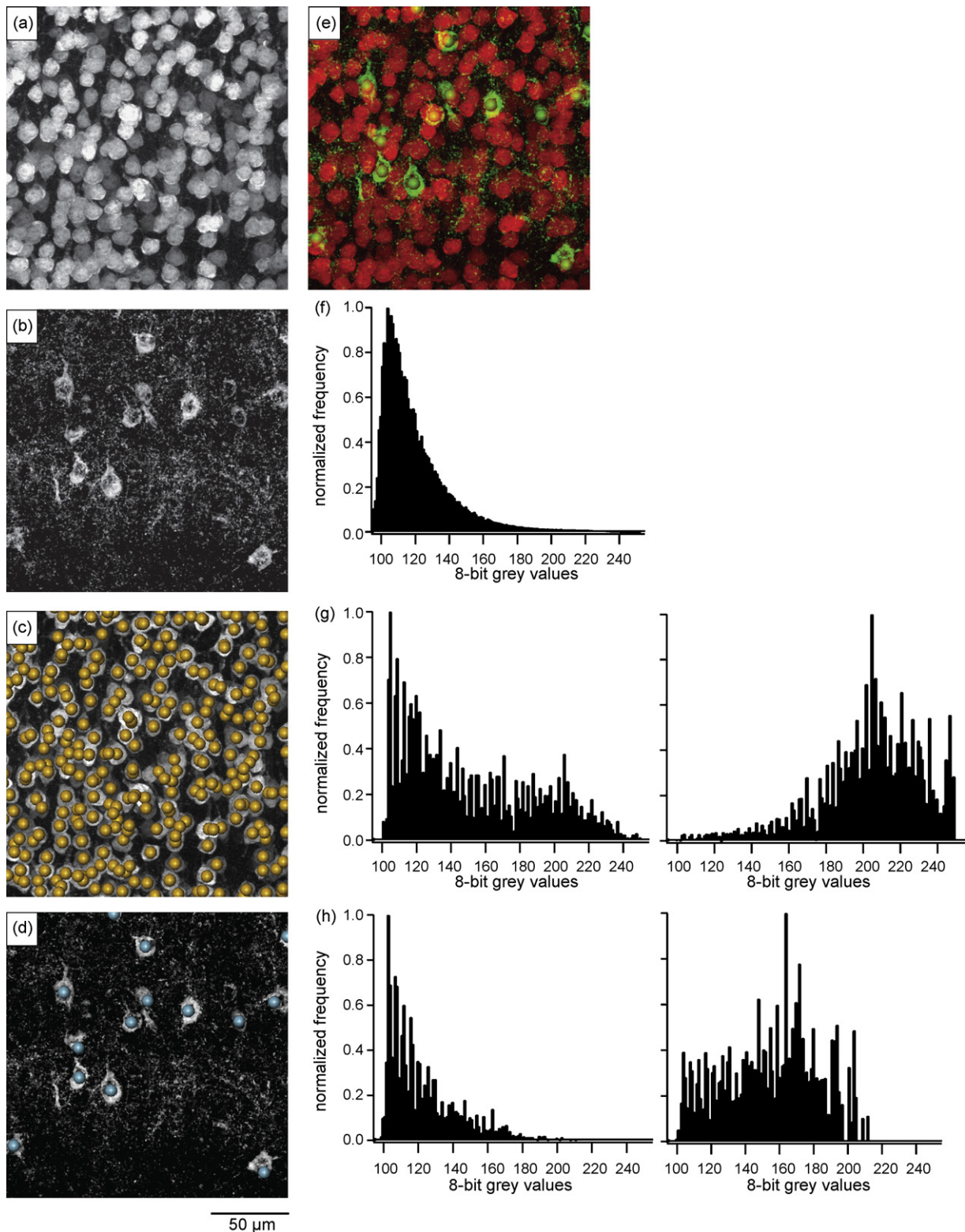
Fig. 10. (a) Maximum x/z-projection of sub-volume of NeuN-stained confocal image stack. (b) Maximum x/z-projection of the same sub-volume of GAD67 counterstained confocal image stack. (c) Automatically detected landmarks of the NeuN stack. (d) GAD67 positive landmarks generated by evaluating the surrounding of each NeuN landmark in the pre-processed version of (b), (e) overlay of the two projections from (a) and (b) with the automatically detected GAD67 positive landmarks. Green reflects neurons stained in both channels. Each green object is reliably labeled by a GAD67 landmark. The accuracy of the second channel GAD67 detection is dependent on the accuracy of the NeuN detection. (f) Normalized gray value histogram of the entire GAD67 image stack. (g) Left: normalized gray value histogram in the GAD67 stack around a NeuN landmark position. Right: same histogram as before, now divided by "background" histogram from (f); due to the presence of a GAD67-positive neuron, high gray values are dominant (h); left: normalized gray value histogram in the GAD67 stack around a NeuN landmark position. Right: same histogram as before, now divided by "background" histogram from (f); due to the absence of a GAD67-positive neuron, low gray values are dominant → if the divided histogram is dominated by gray values below 200 no GAD67 landmark is assigned, if it is dominated by gray values higher than 200, a co-localized GAD67 landmark is set.

to the original image stacks. This effect is also compensated by the described correction filter.

2.7. Multi-channel images

NeuN reliably labels all neuron somata within an image (Gittins and Harrison, 2004). It is hence possible to counter-stain specific neuron types and to image them simultaneously at a different fluorescent wavelength, acquiring so called multi-channel image stacks. Since all neurons within the NeuN channel are detected by the prior described pipeline, the local surrounding of each landmark voxel is inspected in the other channels. If a neuron is also detected in one of the other channels, not only the position and density of all neurons within the stack can be evaluated, but also the neuron type can be obtained.

Fig. 10 shows an example for this procedure. GAD67 positive interneurons were co-localized in a second fluorescent channel (Fig. 10a/b). The pre-processing of the second channel is again dependent on the staining and imaging techniques. In the case of GAD67, imaged by a confocal laser scanning microscope, immunoreactive GABAergic interneurons show a strong somatic GAD67-signal (Kaufman et al., 1991; Meyer et al., in preparation-b). However, GAD67 positive synapses will also be stained, resulting in small (<10 voxel) bright spots (Fig. 10b). Hence, the major step of the pre-processing of the GAD67 channel is to get rid of these synapses.

Here the pre-processing is based upon a gradient magnitude filter (Gonzalez and Woods, 2002) as implemented by the ITK (Ibanez et al., 2005). The resultant gradient image is further subjected to an exponential intensity mapping using the following equation:

$$I' = \exp\left(\frac{\log(256)}{255} \times I\right) - 1, \tag{8}$$

where I' and I refer to the intensity (gray) values after and before the mapping, respectively. This mapping effectively attenuates low background values and enhances the synaptic signals. The result is then subtracted from the original image. After these additional GAD67 specific filters, the brick-wise intensity mapping as described in Section 2.4.2 is applied. A final median filter with a structuring element of 1 pixel radius reduces any remaining small artifacts, leaving only GAD67-positive neuron somata (Fig. 10d).

Fig. 10f shows a histogram of such a pre-processed stack. Since the fraction of GAD67-positive neurons among all neurons is low (~15%) (Beaulieu, 1993), the stack histogram is dominated by the background. Fig. 10g/h show intensity histograms around landmark positions that were generated for the NeuN channel images. Fig. 10g (left) shows the landmark histogram of the k-mean cluster around the corresponding NeuN landmark in the presence of a GAD67-positive neuron, whereas the k-means cluster of the landmark used for the landmark histogram in Fig. 10h (left) does not contain such a neuron. All histograms are normalized to one and landmark histograms are shifted so that the bin with the maximum number of gray values coincides with the maximal bin of the stack histogram. These matched histograms are bin-wise divided and the resultant ratio histogram is again normalized to one (Fig. 10g/h, right).

If no GAD67-positive neuron is present at the landmark position, the landmark histogram resembles the shape of the stack histogram, since this is dominated by background. Therefore the divided histogram is more or less flat and shows no contribution at high gray values that correspond to foreground (Fig. 10h, right). In contrast, if a GAD67-positive neuron is present, the amount of foreground pixels is overrepresented in the landmark histogram with respect to the stack histogram. Hence, the ratio histogram shows a significant distribution at high gray values (Fig. 10g, right). The criterion for co-localization can therefore be defined as a threshold function. If 20% of the object's pixels fall into bins from 200 to 255 of

the ratio histogram the neuron is considered to be present in both channels.

3. Results

3.1. Detection accuracy for confocal stacks of NeuN/GAD67-stained neurons

The evaluation of the automated counting pipeline is done for eleven confocal image stacks of NeuN/GAD67-stained neuron somata. In general, the scientific interest focuses on neuron densities. Hence the absolute number of neurons within the stack volume and the deviation of the neurons' position from manual counts need to be investigated. For this error analysis it is necessary to determine the corresponding landmarks from the manually and automatically processed sets. If the deviation in position between the automated and manually detected landmarks is of negligible amount, the absolute difference in landmark numbers can be regarded as a meaningful error value. In addition, the numbers of FP and FN landmarks are checked. We use manually determined landmarks as the "gold standard" and compare them with our automated counting results. It should however be emphasized that the results of the manual counting are only accurate within 2.1% precision (Meyer et al., in preparation-b).

The eleven image stacks were randomly chosen from a large data pool that was used for evaluation of neuron densities within a cortical column of S1 in rats (Meyer et al., in preparation-a). The stacks were taken at various magnifications ($40\times$, $63\times$) resulting in different resolutions and stack volumes, at various cortical depths (300–$1800\,\mu m$ from the pia surface) yielding different dominant neuron populations in each sample (e.g. layer 4 spiny stellates or layer 5 pyramidal neurons) (Schubert et al., 2007) and at various slice orientations (i.e. thalamocortical, coronal, tangential). Further the comparison was performed "double blind", meaning stacks were first evaluated manually by different individuals and afterwards processed by the automated pipeline described above, without prior inspection of the manual counts.

3.1.1. Counting difference

Table 1 shows the results for the comparison of automatically and manually detected landmarks. It is important to notice that no systematic miscounting is performed by the automated pipeline. The number of detected neurons differs similarly in both directions resulting in an average counting difference that is less than 1% (0.92% averaged over the number of data sets). However, the average absolute counting difference in landmark numbers is around 4%. This value is regarded as the absolute error of the automated counting.

3.1.2. Deviation in landmark position

The average radius of neurons in S1 is between 5 and 15 μm. If the deviation between a manually placed landmark and its automated counterpart is sufficiently lower than the smallest radius value of 5 μm, the error in position can be neglected and the absolute difference in landmark numbers is regarded as a meaningful error value.

For evaluation of position deviation the following algorithm was used:

1. Calculation of distances from each automated landmark to each manual landmark.
2. Sorting of distances, starting with the shortest one and assignment of this manual landmark as the nearest neighbor (NN) of the automated one.
3. Checking for multiple times assigned NN. If more than one automated landmark has the same NN, the closest automated one

will keep this NN, the other automated landmarks are assigned to have their second closest manual landmark as their NN.

4. Step 3 is repeated until no manual landmark is assigned to more than one automated landmark.
5. The distance of all automated landmarks to their assigned manual NN is then averaged (see Table 1).

The average deviation in position of $3.37 \pm 1.11\,\mu m$ is sufficiently less than $5\,\mu m$. It is further important to notice that the position deviation for each dataset is smaller than $5\,\mu m$. Hence the error in position can be neglected, because each automated landmark has a manual counterpart within reasonable distance. It is therefore justified to state that the absolute difference between manually and automatically detected neuron somata is 4%.

3.1.3. False positive/negative landmarks

In addition to the above considerations about average differences between manual and automated counts, we performed a false positive/negative analysis for five randomly chosen stacks. We visualized the two corresponding landmark sets in Amira and manually deleted corresponding landmarks. The remaining automated landmarks were regarded as FP and the remaining manual landmarks were regarded as FN. These FP/FN objects are usually ambiguous cases of touching neurons that could for instance either be counted as one or two neuron somata. Table 2 shows the results for this analysis. Both, the average FP and FN values are around 5%. This compensatory effect explains the average relative counting difference of less than 1%.

3.1.4. Detection of GAD67 neurons from 2-channel images

The detection of neurons in the second channel is strongly dependent on the performance of the detection in the first channel. If the neuron is identified correctly in the first (e.g. NeuN) channel, the detection in the second (e.g. GAD67) channel was 100% correct. We can hence transfer the absolute error value of 4% to the detection of GAD67 positive interneurons.

3.2. Adaptation to different imaging systems, stains and magnification

The analysis performed for the confocal NeuN/GAD67 stacks was applied to two *in vivo* 2p image stacks of Ca^{2+}-sensitive neurons. Table 1 shows that the average differences in numbers of automatically and manually detected somata are of similar amount as in the confocal case. In contrast, the deviation in position seems to be almost twice as large compared to the NeuN/GAD67 images. However, we argue that this position deviation is still negligible. The magnification of the 2p images is more than 2 times worse along each spatial direction (Gobel et al., 2007). Hence, the manual assignment of a landmark is much more imprecise. If the landmark positions differ by just one pixel, the deviation is maximally

Table 2
False positive landmarks (FP), automated neuron position references with no manual counterparts and false negative landmarks (FN), manual position references with no automated counterparts were investigated for 5 confocal NeuN image stacks. Both values are similar and therefore compensate each other, explaining the low values of relative neuron number differences.

	FN [%]	FP [%]
NeuN confocal 40×	9.30	2.78
	7.23	6.31
	4.71	7.58
	4.65	5.35
	3.19	3.01
Mean	5.82	5.01
Std	2.43	2.08

$0.8\,\mu m$ in the confocal case ($40\times$), but already $1.67\,\mu m$ for the 2p images, partly explaining the 2-fold increase in position deviation. We therefore conclude that the automated pipeline performs with similar accuracy for confocal stacks of NeuN/GAD67 stained somata and 2p stacks of Ca^{2+}-sensitive neurons.

Resolution and SNR after deconvolution of NeuN somata from fluorescent widefield image stacks are similar to NeuN/confocal stacks. Hence, counting results and performance, when compared to manual counts, will be similar for both imaging techniques (Table 1).

4. Conclusion

We presented a novel approach for fast automated detection of stained neuron somata on the example of NeuN/GAD67 confocal image stacks. The processing pipeline is based upon three subsequent steps: threshold-based pre-processing, watershed-based morphological filtering and model-based correction algorithms. By adjusting the pre-processing step, it was shown that the pipeline can be adapted to deconvolved widefield stacks of NeuN labeled neurons and to *in vivo* 2p images of Ca^{2+}-sensitive neurons. Recently, we applied the neuron counting pipeline described above to confocal mosaic image stacks of NeuN labeled neurons in the bulbus olfactorious of mice (personal communication Andreas Schaefer, Jan Herb, Max Planck Institute of Medical Research, Heidelberg, Germany) and the ventral-posterior-medial (VPM) nucleus in the thalamus of rats. In consequence, we argue that our approach is applicable to many brain regions of different species.

By comparing the automated results with manually generated counterparts and regarding these manual counts as the "golden standard" result (i.e. neglecting the 2.1% inter-user-variability), the automated system correctly reproduces manually detected neuron somata with more than 90% accuracy. Compensatory effects of approximately 5% FP as well as FN detections from ambiguous clusters of neuron somata result in average relative counting differences of less than 1%. The average error in absolute numbers is around 4%. High accuracy of the automated counting of NeuN-stained somata allowed for detection of co-localized GAD67 positive interneurons from multi-channel image stacks. In consequence the presented system will not only allow for a quantitative analysis of neuron density gradients across or between anatomical/functional subunits, but might even distinguish between different neuron types from multi-channel image stacks.

However, since the last processing step is model-based and hence restricted by constraints, the approach presented here has certain limitations. We assume that a single dominating neuron population with Gaussian-distributed volumes is present within each field of view (1024×1024 pixel wide image stack). If a second minor population of large neurons is present within the image, it is assumed to be spatially separated from the dominant population. In summary, the first mode of the watershed object volume histogram should not be blurred by a second or third population. These constraints are met for the presented "small" image size of $375\,\mu m \times 375\,\mu m \times 50\,\mu m$ for most brain regions. Nevertheless, isolated large neurons (e.g. in Layer 5B of S1) can lead to a significant overestimation of neuron density (personal communication with Hanno Sebastian Meyer, MPI for Medical Research, Heidelberg, Germany). At this stage these singular events are corrected by manually editing the landmark file and superimposing the original image stack in Amira.

Future improvements could incorporate more constraints about neuron shape, size or expected number of neuron populations within the inspected volume. For instance evaluating the volume histogram after the splitting step will yield more accurate neuron volumes and potentially differentiate between sub-populations

of neurons. However, further constraints strongly depend on the kind of data and are unnecessary for the presented data and hence beyond scope of this article.

Further we would like to argue that the above described method is not only more accurate than classical sparse sampling methods, but will reveal new insights in brain organization and will hence produce deviating results from previous studies. By investigating a volume of ∼8 mm³ of S1 in rats we observe significant density variations along all three axes within each cortical layer (preliminary results). Hence, sparse sampling methods will yield strongly deviating results, depending on the sample position and orientation within the cortex. By counting the absolute neuron number and hence obtaining the 3D neuron distribution across the entire area of interest, our method is independent of sample orientation, position or shrinkage and will potentially yield results (Meyer et al., in preparation-b) differing from previously reported studies (Beaulieu, 1993).

In summary, the here presented counting pipeline is a fast (∼1 h computing time per stack), reliable and objective alternative to manual counting of neurons. It is applicable to different brain areas, yields absolute numbers and 3D neuron distributions. It is therefore possible to obtain statistically valid neuron densities for functional subunits in the brain, revealing their detailed anatomical basis.

Acknowledgements

References

Abercrombie M. Estimation of nuclear population from microtome sections. AnatRec 1946;94:239–47.

Agard DA. Optical sectioning microscopy: cellular architecture in three dimensions. Annu Rev Biophys Bioeng 1984;13:191–219.

Akemann W, Zhong YM, Ichinohe N, Rockland KS, Knopfel T. Transgenic mice expressing a fluorescent in vivo label in a distinct subpopulation of neocortical layer 5 pyramidal cells. J Comp Neurol 2004;480:72–88.

Amos WB, White JG, Fordham M. Use of confocal imaging in the study of biological structures. Appl Opt 1987;26:3239–43.

Beaulieu C. Numerical data on neocortical neurons in adult rat, with special reference to the GABA population. Brain Res 1993;609:284–92.

Chang H, Parvin B. Segmentation of three dimensional cell culture models from a single focal plane. Berlin/Heidelberg: Springer; 2006.

Cragg BG. The density of synapses and neurones in the motor and visual areas of the cerebral cortex. J Anat 1967;101:639–54.

Donaldson HH. Growth of the brain: study of nervous system in relation to education. New York: Scribner; 1895.

Gittins R, Harrison PJ. Neuronal density, size and shape in the human anterior cingulate cortex: a comparison of Nissl and NeuN staining. Brain Res Bull 2004;63:155–60.

Gobel W, Kampa BM, Helmchen F. Imaging cellular network dynamics in three dimensions using fast 3D laser scanning. Nat Math 2007;4:73–9.

Gonzalez R, Woods R. Digital image processing. second ed. Upper Saddle River, NJ 07458: Prentice-Hall, Inc.; 2002.

Helmstaedter M, de Kick CP, Feldmeyer D, Bruno RM, Sakmann B. Reconstruction of an average cortical column in silico. Brain Res Rev 2007;55:193–203.

Ibanez L, Schroeder W, Ng L, Cates J. The ITK software guide; 2005, http://www.itk.org.

Kaufman DL, Houser CR, Tobin AJ. Two forms of the gamma-aminobutyric acid synthetic enzyme glutamate decarboxylase have distinct intraneuronal distributions and cofactor interactions. J Neurochem 1991;56:720–3.

Knox CA. Effects of aging and chronic arterial hypertension on the cell populations in the neocortex and archicortex of the rat. Acta Neuropathol 1982;56:139–45.

Kumar SS, Buckmaster PS. Neuron-specific nuclear antigen NeuN is not detectable in gerbil subtantia nigra pars reticulata. Brain Res 2007;1142:54–60.

Li K, Miller ED, Weiss LE, Campbell PG, Kanade T. Online tracking of migrating and proliferating cells imaged with phase-contrast microscopy. In: Conference on computer vision and pattern recognition workshop; 2006.

Lin G, Adiga U, Olson K, Guzowski JF, Barnes CA, Roysam B. A hybrid 3D watershed algorithm incorporating gradient cues and object models for automatic segmentation of nuclei in confocal image stacks. Cytometry Part A 2003;56A:23–36.

Lin G, Chawla MK, Olson K, Barnes CA, Guzowski JF, Bjornsson C, et al. A multi-model approach to simultaneous segmentation and classification of heterogeneous populations of cell nuclei in 3D Confocal microscope images. Cytometry Part A 2007;71A:724–36.

Lin G, Chawla MK, Olson K, Guzowski JF, Barnes CA, Roysam B. Hierarchical, model-based merging of multiple fragments for improved three-dimensional segmentation of nuclei. Cytometry Part A 2005;63A:20–33.

Liu TM, Li G, Nie JX, Tarokh A, Zhou XB, Guo L, et al. An automated method for cell detection in zebrafish. Neuroinfo 2008;6:5–21.

Luo L, Callaway EM, Svoboda K. Genetic dissection of neural circuits. Neuron 2008;57:634–60.

Malpica N, deSolorzano CO, Vaquero JJ, Santos A, Vallcorba I, GarciaSagredo JM, et al. Applying watershed algorithms to the segmentation of clustered nuclei. Cytometry 1997;28:289–97.

Mercury-Computer-Systems. Amira; 2007, http://www.amiravis.com/.

Meyer HS, Wimmer VC, Oberlaender M, Sakmann B, De Kock CP, Helmstaedter M. Neuron density profiles of a thalamocortical innervation column in rat somatosensory cortex. in preparation-a.

Meyer HS, Wimmer VC, Oberlaender M, Sakmann B, Helmstaedter M. The number and distribution of interneurons in a cortical column of rat barrel cortex; in preparation-b.

Mullen RJ, Buck CR, Smith AM. Neun, a neuronal specific nuclear-protein in vertebrates. Development 1992;116:201–11.

Muzio L, DiBenedetto B, Stoykova A, Boncinelli E, Gruss P, Mallamaci A. Conversion of cerebral cortex into basal ganglia in Emx2(−/−) Pax6(Sey/Sey) double-mutant mice. Nat Neurosci 2002;5:737–45.

Nilsson B, Heyden A. Segmentation of complex cell clusters in microscopic images: application to bone marrow samples. Cytometry Part A 2005;66A:24–31.

Oberlaender M, Broser PJ, Sakmann B, Hippler S. Shack–Hartmann wavefront measurements in cortical tissue for deconvolution of large three-dimensional mosaic transmitted light brightfield micrographs. J Microsc 2009;233:275–89.

Oberlaender M, Bruno RM, Sakmann B, Broser PJ. Transmitted light brightfield mosaic microscopy for three-dimensional tracing of single neuron morphology. J Biomed Opt 2007;12.

Peters A, Kara DA, Harriman KM. The neuronal composition of area-17 of rat visual-cortex. 3. Numerical considerations. J Comp Neurol 1985;238:263–74.

Raman S, Maxwell CA, Barcellos-Hoff MH, Parvin B. Geometric approach to segmentation and protein localization in cell culture assays. J Microsc 2007;225: 22–30.

Ranzato M, Taylor PE, House JM, Flagan RC, LeCun Y, Perona P. Automatic recognition of biological particles in microscopic images. Pattern Recogn Lett 2007;28: 31–9.

Rockel AJ, Hiorns RW, Powell TP. The basic uniformity in structure of the neocortex. Brain 1980;103:221–44.

Schubert D, Kotter R, Staiger JF. Mapping functional connectivity in barrel-related columns reveals layer- and cell type-specific microcircuits. Brain Struct Funct 2007;212:107–19.

Scientific-Volume-Imaging. Huygens professional; 1995–2007, http://www.svi.nl.

Soille P. Morphological image analysis, principles and applications. Berlin/Heidelberg: Springer-Verlag; 1999.

Staiger JF, Masanneck C, Bisler S, Schleicher A, Zuschratter W, Zilles K. Excitatory and inhibitory neurons express c-Fos in barrel-related columns after exploration of a novel environment. Neuroscience 2002;109:687–99.

Stalling D, Westerhoff M, Hege H. Amira: a highly interactive system for visual data analysis. In: Hansen CD, Johnson CR, editors. The visualisation handbook. Elsevier; 2005. p. 749–67.

Sterio DC. The unbiased estimation of number and sizes of arbitrary particles using the disector. J Microsc 1984;134:127–36.

Stroustrup B. The C++ programming language. third ed. Addison-Wesley; 2004.

Vincent L. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. IEEE Trans Image Process 1993:176–201.

Vincent L, Dougherty ER. Morphological segmentation for textures and particles. In: Dougherty ER, editor. Digital image processing: fundamentals and applications. Marcel-Dekker; 1994. p. 43–102.

Vincent L, Soille P. Watersheds in Digital Spaces—an efficient algorithm based on immersion simulations. IEEE Trans Pattern Anal Mach Intell 1991;13:583–98.

Warren MA, Bedi KS. A quantitative assessment of the development of synapses and neurons in the visual cortex of control and undernourished rats. J Comp Neurol 1984;227:104–8.

Werner L, Wilke A, Blodner R, Winkelmann E, Brauer K. Topographical distribution of neuronal types in the albino rat's area 17. A qualitative and quantitative Nissl study. Z Mikrosk Anat Forsch 1982;96:433–53.

Williams RW, Herrup K. The control of neuron number. Ann Rev Neurosci 1988;11:423–53.

Williams RW, Rakic P. 3-Dimensional counting—an accurate and direct method to estimate numbers of cells in sectioned material. J Comp Neurol 1988;278:344–52.

Wolf HK, Buslei R, SchmidtKastner R, SchmidtKastner PK, Pietsch T, Wiestler OD, et al. NeuN: a useful neuronal marker for diagnostic histopathology. J Histochem Cytochem 1996;44:1167–71.

Wu HS, Barba J, Gil J. Iterative thresholding for segmentation of cells from noisy images. J Microsc 2000;197:296–304.

Wu KN, Gauthier D, Levine MD. Live cell image segmentation. IEEE Trans Biomed Eng 1995;42:1–12.