Development of deep learning models for microglia analyses in brain tissue using DeePathology™ STUDIO

Luisa Möhle a,*, Pablo Bascuñan a, Mirjam Brackhan a,b, Jens Pahnke a,b,c,*

a Department of Pathology, Translational Neurodegeneration Research and Neuropathology Lab, University of Oslo (UiO) and Oslo University Hospital (OUS), Oslo, Norway
b LIED, University of Lübeck, Lübeck, Germany
c Department of Pharmacology, Faculty of Medicine, University of Latvia, Riga, Latvia

ARTICLE INFO

Keywords:
Artificial intelligence
Deep learning algorithm
Histology
Image analysis
Machine learning
Microglia
Alzheimer

ABSTRACT

Background: Interest in artificial intelligence-driven analysis of medical images has seen a steep increase in recent years. Thus, our paper aims to promote and facilitate the use of this state-of-the-art technology to fellow researchers and clinicians.

New method: We present custom deep learning models generated in DeePathology™ STUDIO without the need for background knowledge in deep learning and computer science underlined by practical suggestions.

Results: We describe the general workflow in this commercially available software and present three real-world examples how to detect microglia on IBA1-stained mouse brain sections including their differences, validation results and analysis of a sample slide.

Comparison with existing methods: Deep-learning assisted analysis of histological images is faster than classical analysis methods, and offers a wide variety of detection possibilities that are not available using methods based on staining intensity.

Conclusions: Reduced researcher bias, increased speed and extended possibilities make deep-learning assisted analysis of histological images superior to traditional analysis methods for histological images.

1. Introduction

Evaluation of histological sections is a core practise in clinical and research routine. In recent years, interest in deep learning-assisted histological analysis has spiked from around 20–50 publications per year (1975–2015) to 300–600 (2018–2020). This reflects, among other factors, changes in availability and technological advances (i.e. faster computers) that make deep learning more accessible.

Deep learning has clear advantages over conventional methods. It can save 90% of analysis time (Bascunana et al., 2021) and improves detection sensitivity (Klein et al., 2020). However, developing deep learning algorithms often involves computer scientists with knowledge in the field of artificial intelligence (AI). Instead, we used DeePathology™ STUDIO (further referred to as STUDIO), which is a commercially available do-it-yourself platform to develop custom deep learning-based models without the need to write computer code. Slides are stored locally and do not need to be uploaded to an external third-party server.

We have shown recently that STUDIO is an appropriate tool to produce results with high quality and improved reproducibility between individual researchers. Number and size of β-amyloid plaques and activated microglia determined with deep learning models correlated closely with values obtained from our classic method of object segmentation with AxioVision Software (Carl Zeiss Microscopy Deutschland GmbH, Jena, Germany) (Bascunana et al., 2021). Additionally, deep learning opens the door for refined object classification (Bascunana et al., 2021). While our previous publication focusses on the comparison of two different analysis methods, it does not go into detail about how...
the models were trained.

In this follow-up report, we provide a comprehensive guide on how to successfully develop custom models, particularly using the versatile Objects mode of STUDIO. Despite being easy to use, this new tool may seem intimidating to users, especially if not familiar with AI-assisted analysis. We give practical suggestions on how to start and what to be aware of during the process. Moreover, we demonstrate the vast possibilities of deep learning models by comparing three different models developed with the same staining and tissue type.

2. Material & methods

2.1. Immunohistology

Formalin-fixed hemispheres from APPtg mice (amyloid precursor protein (APP) and presenilin 1 (PS1) double-transgenic mice, strain APPPS1–21 (Radde et al., 2006)) were embedded in paraffin and cut in 4 μm thick coronal sections. Sections (approximately Bregma ~2.0 mm) were stained for microglia [ionized calcium-binding adapter molecule 1 (IBA1), 1:1000, FUJIFILM Wako Chemicals Europe GmbH, Neuss, Germany, 019-19741] using a BOND-III® automated immunostaining system (Leica Biosystems GmbH, Nussloch, Germany) with a haematoxylin counterstain (provided with the staining kit). After staining, tissue sections were digitized at 230 nm resolution using a Pannoramic MIDI II slide scanner (3DHISTECH Ltd., Budapest, Hungary) (Frohlich et al., 2016; Hofrichter et al., 2016; Krohn et al., 2015, 2011; Paarmann et al., 2019; Steffen et al., 2016, 2017).

2.2. Hard- and software

We used DeePathology™ STUDIO (DeePathology Ltd., Ra’anana, Israel) in the April-2021-version on a virtual machine with Windows 10 Enterprise N, two Intel® Xeon® Gold 6154 processors, an NVIDIA GRID RTX6000P–24Q graphics card and 32 GB random-access memory (RAM).

3. Deep learning model creation

3.1. Project initiation

Currently, STUDIO features four different modes that use different neural networks as a basis:

Regions: Ideal for regional segmentation, i.e. classification of large areas into different categories based on their cellular pattern.

Cells: Best suited to detect and classify cells with typical round/oval shape and smaller size. Works with isolated cells (e.g. blood smears or cytospins) and single cells in tissue structures, e.g. detecting NeuN-positive neurons. Less suitable to detect non-round shaped cells like microglia or astrocytes.

Objects: Versatile approach that uses instance segmentation neural networks and excels when identifying different cells within a tissue or other objects such as plaques. Objects can have any shape or size. Less suitable to detect many distinct objects close to each other in dense areas (in this case, “cells” may be better suited).

Tiles: Used for analysis of larger fields for pattern/tissue recognition. The given area is divided into squares and each square is classified into a predetermined category.

We used the “Objects” mode to develop our microglia-detecting algorithms.

3.2. Establishing categories and settings

After the project has been initiated, one or more slides were added to the project. These slides contained representative examples of all categories of interest and were similar to the slides to be analysed.

First, we added all categories that should be included in the model. Category 1 is always background, i.e. everything that is not supposed to be detected as object.

Note: Adding or removing categories later on will reset the model and training will start from zero.

The user can choose the most suitable settings for the algorithm now (see Table 1), but all settings (besides categories) can also be changed and adapted later on if needed. Next, we chose 5–10 perfect examples of objects belonging to each category (including background), marked them and assigned them to the correct category. When annotating an object, we did not include any other objects in the selection, even if they were overlapping with the desired object.

Note: When thinking about categories, users should have a precise picture in mind what objects in each category look like. Training an algorithm requires clear and unambiguous input – if you are confused, the algorithm will give confusing results.

Choosing the background was particularly important: we selected different areas containing objects that were not supposed to be detected, not only empty areas commonly referred to as “background”.

3.3. Initiating learning and monitoring progress

As soon as we had selected enough objects per category (at least 5), STUDIO started generating an algorithm from the examples. The ongoing learning process was indicated by “training in action” in the menu bar.

We monitored the learning process under “Graphs”. In Objects mode, the graph showed the loss function used for training the neural network, i.e. “epochs” on the X-axis and “training loss” on the Y-axis (Fig. 1). One epoch means that each annotated sample has been included in forming the algorithm. The time required for each epoch depends largely on the computational power of the machine and the number of objects included in the training set. Training loss, the other parameter in the chart, is an indicator for the accuracy of the algorithm’s predictions. Zero represents perfect predictions and the higher the training loss the less accurate the prediction is. After we corrected the algorithm (i.e. changing categories of detected objects or adding new objects), training loss numbers increased until the algorithm had adapted to include the new data.

As time passed, the training dataset was analysed repeatedly. The epoch number increased and the training loss generally decreased (as long as we did not give new input), indicating improvement of the algorithm.

3.4. Increasing training dataset

When the basic object recognition worked, the training dataset needed to be increased to ensure broad applicability. At this point, images/scans with varying staining intensity were added to include objects from different slides in the training dataset.

We employed multiple ways to increase the training dataset: We added new objects manually, through test runs and inside the gallery.

Manually: We added new objects at any point by drawing the outline manually. This was the slowest method, but at the same time maybe the most accurate, as we had full control over the exact outline. This method was most relevant in the beginning of the training phase, when the algorithm’s predictions were not yet accurate enough. However, it also became relevant later on. We noticed that while the algorithm was recognizing the correct objects, it did not trace them as accurately as we did as humans (Fig. 2). With the addition of less accurately outlined objects and the subsequent changes in the algorithm, the margin between the desired outline and the detected outline got larger, especially when we detected objects with a complex shape (i.e. not round). In this situation, we added manually outlined objects from time to time to improve detection.
Test runs (detection-evaluation process): We used preliminary versions of the algorithm to run it on specific areas of the slide, where it outlined predicted objects in the current field of view. We then added

### Table 1
Customization parameters in DeePathology™ STUDIO.

| Parameter | Options (default) | Explanation |
|-----------|------------------|-------------|
| Area Threshold (in pixels) | 1–20,000px (2px) | Minimum size a predicted object has to have in order to be counted/displayed as detected object. This setting does not affect training and can be changed at any time. The size of any predicted object can be seen in the “Viewer” tab. Note that the conversion factor between pixels and µm² depends on the image resolution. |
| Use Augmentation | Yes/No | Yes (recommended): The AI uses copies of existing objects that have been slightly changed (e.g., rotated, flipped, scaled, etc.) to increase the training dataset and reduce overfitting. Additionally, this can prevent the AI from focusing on irrelevant patterns and thus, improve performance. |
| Freeze Status | Running/Frozen | Running: Training mode. Learning is activated and the algorithm changes with new input. Frozen: Analysis mode. Learning is not active, the algorithm is preserved. |
| Import Segmentation/Tiles Project | None/Imported project | Import a previously generated project that contains an algorithm to detect regions or classify tiles to cross-analyse, e.g. objects per region. |
| Resolution Level | 0 (high resolution)-9 (low resolution) | Lower resolution levels result in faster analysis, but are not suitable for objects with very fine details. Lower resolutions are often recommended for region analysis. |
| Use Tissue Detection | Yes/No | Yes: White areas without tissue are automatically excluded. No: Areas without tissue are included. |
| Model version to use | Latest/Best | Latest: Uses the latest version of the algorithm to detect objects. Best: Uses the best version of the algorithm as determined by training accuracy values. |
| Active learning buffer | 1 - x seconds (180 s) | Maximum age of the objects displayed in the gallery. |
| Generate raw output data in reports | Yes/No | Yes: When generating a report by analysing a region of interest (ROI), the Excel file will contain an additional tab that lists all detected objects including X/Y coordinates, category and area. No: Only the summary is saved as a report. |
| Export Detected Boundaries as XML in reports | Yes/No | Yes: When generating a report by analysing an ROI, an additional XML file is created that contains precise outline coordinates for all detected objects. No: No XML file is generated. |
| % of ROIs to save as screenshots | 0–100% (20%) | When generating a report, screenshots of the ROIs are automatically saved to control algorithm performance on a specific slide. This setting determines which percentage of the ROIs are saved as screenshots. If < 100%, tiles are selected randomly. |
| Crop size for screenshots | 64 - 1024px | Determines the area (in pixels) that one screenshot tile covers. Smaller values will give a higher resolution, but increase the total number of image files (with same % settings). |
| Near Duplicate Filtering | Weak/Medium/High | Determines the maximum overlap two predicted objects can have. “Weak” means that objects can overlap significantly, whereas “high” allows only a small or no overlap at all. |

#### Table 1 (continued)

Fig. 1. Monitoring model training progress. The graph shows the training loss for each epoch. Higher values for the training loss indicate less accurate model performance. Blue arrows were added later to indicate time points when new data was added to the training dataset that immediately affected the training loss.

Fig. 2. Manual vs. algorithm-based object outline. The screenshot from DeePathology™ STUDIO shows a side-by-side comparison of a manually outlined object (right) and an object outlined by the algorithm (left). Both objects are part of the training dataset of approach 3 and belong to category 2 “microglia” (see Table 2). The algorithm outlines tended to include more background and to have a rounder shape than the manual outline.

Test runs (detection-evaluation process): We used preliminary versions of the algorithm to run it on specific areas of the slide, where it outlined predicted objects in the current field of view. We then added
one or more of these predicted objects to the training dataset by right-clicking on them. We used the flow chart (detection-evaluation process, Fig. 3) as decision help and made sure to review predicted objects at a relevant magnification. Initially, we chose smaller areas (i.e., larger magnification) for a test run. Once the model’s predictions were more confident, we also analysed larger areas. Importantly, the detection-evaluation process was also performed on “complicated” areas, i.e., areas with different structure and areas of transition (tissue type borders, varying tissue structure, and edges of tissue), dust specs or staining artefacts, tissue folds, etc.

**Gallery:** Inside the gallery, we added suggested objects to any of the categories. We filtered displayed suggestions e.g., by slide, or category, or displayed only “confusing objects”. We used this as a quick way to add many objects to the dataset. If “AI mode” was switched on, predicted objects were automatically classified and added to the respective category. When using “AI mode”, it was crucial to monitor the training dataset in case of false predictions that may flaw the algorithm.

The larger the training dataset became, the more time one epoch lasted. To see the effects of newly added objects on the model, we needed to wait for some epochs after adding the objects until the algorithm had been adapted (i.e., training loss had reduced again). Then, we tested the algorithm again.

### 3.5. Reviewing annotations

We used the “annotation review” feature to review the training dataset, i.e., all annotations or annotations belonging to a specific category. We eliminated incorrectly categorized objects that would confuse the model and may negatively influence its performance. The displayed objects were automatically sorted according to their fit into the assigned category with objects fitting worst (and thus, having the highest chance of being incorrect annotations) appearing on top of the list.

### 3.6. Version control

With most models, we reached a point where we were satisfied with the predictions, but wondered if it could be improved further. We also let the model train over a longer period to see how accurate it could become. In any of those cases, we kept a snapshot of the current status as backup and potential restoration point. A snapshot always contains all information about the current model including object and region annotations, and the current algorithm.

Particularly when training for a longer period, we changed the setting “Model version to use” to “Best” instead of “Latest” to always keep a copy of the until then best model version.

We also used snapshots as templates for new experiments that used the same model (but different slides), and to share a model among colleagues.

### 3.7. Validation

To validate the model, we started by defining a test area. Within this area, we annotated all objects of all categories correctly by hand (except for

---

**Fig. 3.** Detection-evaluation process. The flow chart depicts a possible decision tree whether to add or not to add a predicted object to the training dataset (Section 3.4).
Table 2
Comparison of three different approaches to detect microglia.

| Approach Parameters | (1) Microglia bodies | (2) “Whole” microglia | (3) Microglia + clusters |
|---------------------|----------------------|-----------------------|--------------------------|
| Questions/aim       | How many microglia (absolute or per area)? | How much area do microglia cover? How ramified are microglia? | Are microglia clustering together or remaining separate? How big are the clusters? |
| Objects             | 1. Background        | 1. Background         | 1. Background             |
|                     | 2. Microglial soma   | 2. Whole microglia with visible soma and ramifications | 2. Whole microglia (similar to algorithm) |
|                     |                      |                       | 3. Clusters of activated/dystrophic microglia |
| Object characteristics | Objects are round and clearly distinguishable from background due to staining colour and shape. | Objects have varying, irregular shapes. They share certain similarities with the background. | |
| Complexity          |                      |                      |                          |
| Pitfalls            | The algorithm may detect IBA1-positive cells in the meninges and endothelial cells, or cells with a large, dark blue nucleus (counterstain) if not trained otherwise, especially if staining intensity varies largely between slides. | The user has to decide from the beginning, how much of the ramifications they want to include in the object (e.g. should the algorithm also bridge small gaps or be very conservative). Moreover, the algorithm will initially detect ramifications without (clear) soma as objects. Indecisiveness of the user (when do you count a soma as soma) and variability of microglia shapes and section plane will determine how fast the algorithm improves. Over time, the detected objects will appear more and more round (i.e. more background around the ramifications will be included in the object and more ramifications will be slightly cut at the end). To counteract this effect, manually trace out microglia as needed. | Strictly speaking, “microglia” is a subcategory of “cluster”, as every microglia cluster obviously consists of several microglia. However, this training mainly focused on the difference between resting or moderately activated microglia (category 1) and clusters of highly activated and/or dystrophic microglia (category 2). As microglia activation is not a black-and-white type of status, both categories have similarities and may overlap at times. Thus, high precision is difficult to achieve. |
| Precision           | 0.91                 | 0.61                  | Overall: 0.69             |
|                     |                      |                       | Microglia: 0.69            |
|                     |                      |                       | Cluster: 0.72              |
| Recall              | 0.84                 | 0.99                  | Overall: 0.89             |
|                     |                      |                       | Microglia: 0.90            |
|                     |                      |                       | Cluster: 0.84              |
| Validation details  | Tissue area: 1 087 µm² | True positives: 102 | Tissue area: 1 212 µm² |
|                     | 659 µm²              |                       | True positives: 83         |
|                     | True positives: 138 |                       | (microglia) + 21 (cluster) |
for background). Everything that was not annotated was considered background. We ran the validation to receive statistics on the accuracy of the model (precision and recall). Precision is the rate of true positives among all detected objects. Recall is the rate of true positives among all true annotated objects and is referred to as the model’s sensitivity. STUDIO calculates true/false positives based on the overlap of the predicted objects with the manually annotated objects. The results range from 0 to 1, with higher values indicating better performance.

Validation provides a useful tool to put a model’s performance into numbers. We also used it while still training the model. A low precision value indicated that further training should focus on reducing false positive detections. On the other hand, a low recall value indicated that the algorithm was still missing many objects (false negatives). Frequent validation helped making the training annotations more goal-oriented. However, we did not rely purely on the validation results to judge the algorithm. We compared the validation annotations to the predicted objects, and checked which objects were missed or falsely recognized and whether these mistakes were gross errors or within the margin of error that is to be expected under real-life conditions. Moreover, we decided individually whether we were more willing to accept false positives or false negatives, depending on the purpose of the analysis.

Table 3
DeePathology™ STUDIO settings used for the presented algorithms.

| Parameters                  | Approach (1) Microglia bodies | (2) “Total” microglia | (3) Microglia + clusters |
|-----------------------------|-------------------------------|-----------------------|--------------------------|
| Mode                        | Objects                       | 644px (approx. 39 µm²) | 1px                      | 644px (approx. 39 µm²) |
| Area Threshold              | 644px (approx. 39 µm²)        | 1px                   | 644px (approx. 39 µm²)   |
| Use Augmentation            | Yes                           | 644px (approx. 39 µm²) | 1px                      |
| Freeze Status               | Frozen                        | Frozen                | Frozen                   |
| Resolution Level            | 0 (highest)                   | 0 (highest)           | 0 (highest)              |
| Use Tissue Detection        | Yes                           | Yes                   | Yes                      |
| Model version to use        | Latest                        | Latest                | Latest                   |
| Active learning buffer      | 180 s                         | 180 s                 | 180 s                    |
| Near Duplicate Filtering    | Weak                          | Weak                  | Weak                     |
|                             | Medium                        | Medium                | Medium                   |

Fig. 4. Screenshots of the same slide analysed with three different deep learning algorithms. Cortical region of a coronal brain section stained against IBA1. (A) 2.5x magnification view. Objects with magenta outline are categorized as microglia somas (algorithm 1) or activated microglia (algorithm 2 and 3). Objects outlined in green are microglia clusters (algorithm 3). Black squares indicate the areas selected for higher magnification views. Scale bar represents 500 µm. (B-D) 20x magnification views of the selected areas, analysed with all three different algorithms as in A. All screenshots were taken directly in DeePathology™ STUDIO. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
Training had to be started several times from the beginning because the resulting algorithm did not perform as well as expected. This was usually caused by sub-optimal examples in the training dataset and was hard to correct. Thus, it proved to be the best choice to go back to step one and to make sure that the categories were clearly distinguishable. If the algorithm missed an object that belonged to one of the categories, we annotated the object manually by drawing. When we ran the algorithm again after some time, it usually had improved to include this category besides the background. In contrast, the third approach aimed to recognize two types of objects, ramified microglia and microglia clusters (Fig. 4 A and B). We validated all three models and summarize the validation results in Table 2. Approach 1 (only microglia somas) has a high precision (0.91), indicating few false positives. Slightly lower recall (0.84) suggests a few missed objects (Fig. 4C). Approach 2 (ramified microglia) has an excellent recall value (0.99), but lower precision (0.61), indicating that it has a higher rate of false positives. For approach 3 (ramified microglia + cluster), three values are given for each measure: overall performance and performance in each of the two object categories, the result is nevertheless satisfying. However, further training could be focused on avoiding false positive detection, especially to avoid false detection of microglial processes without corresponding soma (Fig. 4D). We further analysed the same cortical region of interest (area: 10.3 mm²) on the slide shown in Fig. 4. The different object counts reflect the differences in sensitivity and specificity already observed during validation (Fig. 5). The different object counts can be a consequence of the selected target objects and their size, with somas being smaller than microglia including their ramifications, which are in turn smaller than clusters consisting of several microglia cells (Fig. 5).

In some cases, the model started to recognize very small objects. In this case, we used the area threshold setting to filter out unwanted small objects. If we were unsure regarding the threshold value, we investigated the size of relevant detected objects by clicking on them.

Note: This setting does not affect the model itself, the filter is applied only for the detection.

While we generally recommend using perfect examples to train the algorithm, also imperfect examples can contribute to fine-tuning of the algorithm. Similarly, manually classifying “confusing” objects can greatly improve the outcome for difficult decisions.

4. Examples of different algorithms

While it is obvious that different stainings will give different results, also one single staining can be analysed in different ways to answer different questions. Thus, it is crucial that any researcher is aware of their question before devising and training an algorithm. Below, we give three examples of algorithms all developed on brain sections stained for IBA1 against a haematoxylin counterstain. IBA1 has been described as a marker for microglia (Ito et al., 1998), but can also be expressed on (morphologically different) monocytes (Jeong et al., 2013). We chose this staining as it is widely used when studying brain inflammation and neurodegenerative diseases such as Alzheimer’s disease, Parkinson’s disease or Huntington’s disease. For example, we have previously described distinct spatiotemporal microglia activation using similar high-resolution whole-slide imaging in Alzheimer’s disease models (Scheffler et al., 2011).

In Table 2, we compare three different approaches to detect microglia on IBA1 stained mouse brain sections. Fig. 4 shows sample images of the same region analysed with the different models. The main settings used to develop these models in STUDIO are summarized in Table 3.

In the first approach, we aimed for detecting only microglial somas, while the second approach aimed to recognize microglia including their ramifications (Fig. 4 A and B). Both approaches contain only one object category besides the background. In contrast, the third approach aimed to recognize two types of objects, ramified microglia and microglia clusters (Fig. 4 A and B). We validated all three models and summarize the validation results in Table 2. Approach 1 (only microglia somas) has a high precision (0.91), indicating few false positives. Slightly lower recall (0.84) suggests a few missed objects (Fig. 4C). Approach 2 (ramified microglia) has an excellent recall value (0.99), but lower precision (0.61), indicating that it has a higher rate of false positives. For approach 3 (ramified microglia + cluster), three values are given for each measure: overall performance and performance in each of the two categories. While none of the values are outstanding, given the similarities and overlap of the two object categories, the result is nevertheless satisfying. However, further training could be focused on avoiding false positive detection, especially to avoid false detection of microglial processes without corresponding soma (Fig. 4D). We further analysed the same cortical region of interest (area: 10.3 mm²) on the slide shown in Fig. 4. The different object counts reflect the differences in sensitivity and specificity already observed during validation (Fig. 5). The different average object size is a consequence of the selected target objects and their size, with somas being smaller than microglia including their ramifications, which are in turn smaller than clusters consisting of several microglia cells (Fig. 5).
6. Conclusions

Artificial intelligence and deep learning have come a long way from an almost futuristic method accessible only to computer scientists to a common technique that is ready for widespread practical application.

One large area of application is image analysis in clinical diagnostics and biomedical research. If used correctly, employing deep learning algorithms can improve accuracy and efficiency of otherwise lengthy image analysis (Bascunana et al., 2021; Klein et al., 2020). While the theoretical considerations have been discussed elsewhere (Le et al., 2020; Wiestler and Menze, 2020), here we present practical tips and suggestions on how to develop deep learning algorithms in STUDIO through supervised learning — without the need of programming. Supervised learning stands in contrast to unsupervised learning e.g. discovering patient clusters by genetic analysis without manually labelled training data (Lopez et al., 2018). In case of supervised learning, artificial intelligence requires human input: It is the responsibility of human users to select training data and it is their decision when to stop training. The quality of these decisions greatly influences the applicability of any produced algorithm.

We demonstrate our approach through analysis of medical images, specifically histological images, with three practical examples of algorithms analysing the same original slides, but with different aims and complexity. Each approach faced different challenges, but validation revealed satisfying performance. However, these algorithms may be further refined in the future, focussing on their individual weaknesses (false positives and/or false negatives) to improve detection. A recent article with appealing results describes how microglia can be classified into four categories (ramified, rod-like, activated and amoeboid) using a machine learning approach with a convolutional neural network (Lehy et al., 2021). This method requires complex image preparation (including contrast equalization, soma and process detection, cell reconstruction and separation) before the machine learning model can be applied to any region of interest (Lehy et al., 2021). Depending on the application and the experience of the researcher, this complexity may be advantageous or problematic.

Deep learning partially overcomes research bias. First, it does not require the researcher to select features before training, which is an advantage over earlier forms of machine learning (Erickson et al., 2017). Second, model performance is objective, i.e. independent of who uses it once the training phase has concluded. However, the model can become inherently biased if trained with inappropriate data. Indeed, “researcher group” was the biggest contributor (31%) to model performance in a study comparing different models to predict software performance (Shepperd et al., 2014). To this end, the ability to easily share a model with other researchers as included in STUDIO is a first step to develop powerful standardized models for histological image analysis.

5. Discussion

The work of J.P. was supported by the following grants: Deutsche Forschungsgemeinschaft/ Germany (DFG 263024513); Ministerium für Wirtschaft und Wissenschaft Sachsen-Anhalt/ Germany (ZS/2016/05/ 78617); Latvian Council of Science/ Latvia (ShortAbeta Izp–2018–1–0275); Helse Sør-Øst/ Norway (2019054, 2019055); Barnekræftfor- eningen (19008); EEA grants / Norges grants (TACR TARIMAD TO100078); Norges forskningsrådet/ Norway (260786 PROP–AD, 295910 NAPI, 327571 PETABC); European Commission (643417).

PROP–AD and PETABC are EU Joint Programme - Neurodegenerative Disease Research (JPND) projects. PROP–AD is supported through the following funding organisations under the aegis of JPND - www. jpond.eu (A KA #301228 – Finland, BMBF #01ED1605 – Germany, CSO-MOH #30600-12631 – Israel, NFR #260786 – Norway, SRC #2015-06795 – Sweden). PETABC is supported through the following funding organisations under the aegis of JPND - www. jpond.eu (N FR #327571 – Norway, FFR #982717 – Austria, BMBF #01ED2106 – Germany, MSMT #8F21002 – Czech Republic, VIAA # ES RTD/2020/26 – Latvia, ANR #20-JP2W-0002-04 – France, SRC #2020-02905 – Sweden). The projects receive funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement #643417 (JPco-fuND).

CRediT authorship contribution statement

L.M.: Conceived and designed the project, created algorithms, wrote the paper, acquired funding. P.B.: Created algorithms, edited the paper. M.B.: Edited the paper. J.P.: Conceived and designed the project, edited the paper, supervised the project, acquired funding. All authors read and approved the final manuscript.

Competing Interests

L.M. and J.P. are scientific advisors for DeePathology Ltd.

Acknowledgements

We thank Thomas Brüning (University of Oslo) for technical support with histology, and Jacob Gildenblat (DeePathology Ltd.) for technical information about DeePathology™ STUDIO.

References

Asunciana, P., Brackhan, M., Pahnke, J., 2021. Machine learning-supported analyses improve quantitative histological assessments of amyloid-beta deposits and activated microglia. J. Alzheimers Dis. 79, 597–605. https://doi.org/10.3233/JAD-201120.

Erickson, B.J., Korfiatis, P., Akkus, Z., Kline, T.L., 2017. Machine learning for medical imaging. Radiographics 37, 505–515. https://doi.org/10.1148/radiographics.2017160130.

Frohlich, C., Zachwieb, K., Grogger, V., Paarmann, K., Steffen, J., Thurm, C., Schropp, E.M., Bruning, T., Gellerich, F., Radloff, M., Schwabe, R., Lachmann, I., Krohn, M., Ibrahim, S., Pahnke, J., 2016. Activation of mitochondrial complex ii-dependent respiration is beneficial for alpha-synucleinopathies. Mol. Neurobiol. 53, 4728-4744. https://doi.org/10.1007/s12035-015-9999-4.

Hofrichter, J., Krohn, M., Schumacher, T., Lange, C., Feistel, B., Walbroel, B., Pahnke, J., 2016. Sideritis spp. Extracts enhance memory and learning in Alzheimer’s disease. Brain 139, 1-9. https://doi.org/10.1093/brain/awv137.

Jeong, H.K., Ji, K., Min, K., Joe, E.H., 2013. Brain inflammation and microglia: facts and misconceptions. Exp. Neurobiol. 22, 59–67. https://doi.org/10.5607/en.2013.22.2.59.

Klein, S., Gildenblat, J., Ihle, M.A., Merkelbach-Bruse, S., Noh, K.W., Peifer, M., Quass, A., Buttner, R., 2020. Deep learning for sensitive detection of Helicobacter Pylori in gastric biopsies. BMC Gastroenterol. 20, 417. https://doi.org/10.1186/s12876-020-01494-7.

Krohn, M., Bracke, A., Avchalumov, Y., Schumacher, T., Hofrichter, J., Paarmann, K., Frohlich, C., Lange, C., Bruning, T., von Bohlen Und Halbach, O., Pahnke, J., 2015. Accumulation of marine amyloid-beta mimics early Alzheimer’s disease. Brain 138, 2370–2382. https://doi.org/10.1093/brain/awv137.

Krohn, M., Lange, C., Hofrichter, J., Scheffler, K., Stenzel, J., Steffen, J., Schumacher, T., Bruning, T., Plath, A.S., Allen, F., Schmidt, A., Winter, F., Rateitschak, K., Wree, A.,
Gsponer, J., Walker, L.C., Pahnke, J., 2011. Cerebral amyloid-beta proteostasis is regulated by the membrane transport protein ABCC1 in mice. J. Clin. Invest 121, 3924–3931. https://doi.org/10.1172/JCI57867.

Le, W.T., Maleki, F., Romero, F.P., Forghani, R., Kadoury, S., 2020. Overview of machine learning: part 2: deep learning for medical image analysis. Neuroimaging Clin. N. Am. 30, 417–431. https://doi.org/10.1016/j.nic.2020.06.003.

Leyh, J., Paeschke, S., Mages, B., Michalski, D., Nowicki, M., Bechmann, I., Winter, K., 2021. Classification of microglial morphological phenotypes using machine learning. Front Cell Neurosci. 15, 701673 https://doi.org/10.3389/fncel.2021.701673.

Le, W.T., Maleki, F., Romero, F.P., Forghani, R., Kadoury, S., 2020. Overview of machine learning: part 2: deep learning for medical image analysis. Neuroimaging Clin. N. Am. 30, 417–431. https://doi.org/10.1016/j.nic.2020.06.003.

Leyh, J., Paeschke, S., Mages, B., Michalski, D., Nowicki, M., Bechmann, I., Winter, K., 2021. Classification of microglial morphological phenotypes using machine learning. Front Cell Neurosci. 15, 701673 https://doi.org/10.3389/fncel.2021.701673.

Lopez, C., Tucker, S., Salameh, T., Tucker, C., 2018. An unsupervised machine learning method for discovering patient clusters based on genetic signatures. J. Biomed. Inf. 85, 30–39. https://doi.org/10.1016/j.jbi.2018.07.004.

Paarmann, K., Prakash, S.R., Krohn, M., Mobile, L., Brackhan, M., Bruning, T., Eiriz, I., Pahnke, J., 2019. French maritime pine bark treatment decelerates plaque development and improves spatial memory in Alzheimer’s disease mice. Phytomedicine 57, 39–48. https://doi.org/10.1016/j.phymed.2018.11.033.

Radd, R, Bolzoni, T, Kaeser, SA, Coozarasswamy, J, Lindau, D, Stoltze, I, Calhoun, ME, Jaggi, F, Wolburg, H, Gengler, S, Haas, C, Ghetti, B, Czech, C, Holscher, C, Mathews, PM, Jucker, M. 2006. Abeta{42}-driven cerebral amyloidosis in transgenic mice reveals early and robust pathology. EMBO Rep 7, 940–946. http://doi.org/10.1038/sj.embor.7400784.

Scheffler, K., Stenzel, J., Krohn, M., Lange, C., Hofrichter, J., Schumacher, T., Bruning, T., Plath, A.S., Walker, L., Pahnke, J., 2011. Determination of spatial and temporal distribution of microglia by 230nm-high-resolution, high-throughput automated analysis reveals different amyloid plaque populations in an APP/PS1 mouse model of Alzheimer’s disease. Curr. Alzheimer Res. 8, 781–788. https://doi.org/10.2174/156720511797633179.

Shen, D., Wu, G., Suk, H.I., 2017. Deep learning in medical image analysis. Annu Rev. Biomed. Eng. 19, 221–248. https://doi.org/10.1146/annurev-bioeng-071516-044442.

Shepperd, M., Bowes, D., Hall, T., 2014. Researcher bias: the use of machine learning in software defect prediction. IEEE Trans. Softw. Eng. 40, 603–616. https://doi.org/10.1109/tse.2014.2322358.

Steffen, J., Krohn, M., Paarmann, K., Schwitick, C., Bruning, T., Marreiros, R., Muller-Schiffmann, A., Korth, C., Braun, K., Pahnke, J., 2016. Revisiting rodent models: octodon degus as Alzheimer’s disease model? Acta Neuropathol. Commun. 4, 91. https://doi.org/10.1186/s40478-016-0363-y.

Steffen, J., Krohn, M., Schwitick, C., Bruning, T., Paarmann, K., Pietrzik, C.U., Biverstal, H., Jansone, B., Langer, O., Pahnke, J., 2017. Expression of endogenous mouse APP modulates β-amyloid deposition in hAPP-transgenic mice. Acta Neuropathol. Commun. 5, 49. https://doi.org/10.1186/s40478-017-0448-2.

Wiestler, B., Menze, B., 2020. Deep learning for medical image analysis: a brief introduction. Neurooncol. Adv. 2, iv35–iv41. https://doi.org/10.1093/noajnl/vdaa092.