scientific reports

OPEN

EXACT: a collaboration toolset for algorithm-aided annotation of images with annotation version control

Christian Marzahl[1,2✉], Marc Aubreville[1,4], Christof A. Bertram[3], Jennifer Maier[1], Christian Bergler[1], Christine Kröger[2], Jörn Voigt[2], Katharina Breininger[1], Robert Klopfleisch[3] & Andreas Maier[1]

In many research areas, scientific progress is accelerated by multidisciplinary access to image data and their interdisciplinary annotation. However, keeping track of these annotations to ensure a high-quality multi-purpose data set is a challenging and labour intensive task. We developed the open-source online platform EXACT (EXpert Algorithm Collaboration Tool) that enables the collaborative interdisciplinary analysis of images from different domains online and offline. EXACT supports multi-gigapixel medical whole slide images as well as image series with thousands of images. The software utilises a flexible plugin system that can be adapted to diverse applications such as counting mitotic figures with a screening mode, finding false annotations on a novel validation view, or using the latest deep learning image analysis technologies. This is combined with a version control system which makes it possible to keep track of changes in the data sets and, for example, to link the results of deep learning experiments to specific data set versions. EXACT is freely available and has already been successfully applied to a broad range of annotation tasks, including highly diverse applications like deep learning supported cytology scoring, interdisciplinary multi-centre whole slide image tumour annotation, and highly specialised whale sound spectroscopy clustering.

The joint interdisciplinary evaluation of images is critical to scientific progress in many research areas. Specialised interpretation of images strongly benefits from cross-discipline cooperation among experts from different disciplines such as the annotation of pathology microscopy slides with the aim of facilitating routine pathology tasks. The strenuous annotation work can be greatly simplified by customised algorithmic support for medical experts provided by engineers and computer scientists. However, this interdisciplinary cooperation has specific demands on all parties involved. One important aspect to be observed is data privacy and protection. Regulations must be put in place to control who is allowed to access which image set and which data are shared. Furthermore, the tools for viewing and annotating images must be efficient and user-friendly in order to achieve a high level of acceptance among medical professionals. Computer-scientists, however, require traceable high-quality and high-quantity data sets which are essential for reproducibility when creating accurate machine learning algorithms. In order to meet these diverse requirements for annotating image data, a wide variety of open-source software solutions have been designed and published in recent years. These software solutions can be divided into three groups: firstly, offline annotation tools like SlideRunner[1], AnnotatorJ[2], Icy[3], or QuPath[4]. Secondly, web-based solutions focusing on cooperation like Cytomine[5] or OpenHI[6]. And finally, platforms that combine established solutions like Icytomine[7] which combines both Icy and Cytomine. All these solutions support whole slide images (WSIs) and provide open-source access for scientific research purposes.

In the following, we define a set of specific requirements for collaborative annotation software that are—in this combination and at the time of this publication—not satisfied in open-source solutions. Furthermore, we introduce annotation templates and annotation versioning as new requirements.

[1]Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany. [2]Research and Development, EUROIMMUN Medizinische Labordiagnostika AG, Lübeck, Germany. [3]Institute of Veterinary Pathology, Freie Universität Berlin, Berlin, Germany. [4]Faculty of Computer Science, Technische Hochschule Ingolstadt, Ingolstadt, Germany. ✉email: c.marzahl@euroimmun.de

Features	Description
Application	
Online	EXACT is a Django-based server application with a browser client
Cross-platform	The server can be installed on Windows, Linux, Mac and all other systems with Docker support
Multi-center	The multi-center support allows sharing data across multiple institutes with appropriate data privacy management
REST-API	The REST-API supports language independent create, read, update and delete (CRUD) operations on all database fields including image upload and download
Language	The web-server is written in Python while the web-client is HTML and JavaScript based
Plugins	Allow the frontend and backend integration of domain-specific features and analyses
Image-set administration	Combining images to a folder like structure with team access rights and shared annotations scheme
File formats	Images: .tif, .png, bmp, .jpeg, .dcm (partially), .webp
	WSI scanner: .svs, .vms, .vmu, .ndpi, .scn, .mrxs, .iSyntax, .svslide, .bif, .czi, .tif
	Time-Series: .avi, .mkt, .tif
User management	Individual user or group rights for CRUD operations on the database
Annotation	
Types	Box, polygon, line, circle and per image (classification) annotations
Templates	Define a framework for general annotation properties like type (box, polygon etc.), colour and default size
Single click	Single click annotations with background knowledge provied by the templates
Guided screening	A persistent screening mode in a user defined resolution which saves the progress
ML	
Version control	Enables the versioning of image sets with the corresponding image list and annotations
Inference	Performing inference on client side via browser, on server side via Python or over the REST-API

Table 1. Features of EXACT regarding applications, data set annotation and machine learning.

The software should be usable online and offline, while providing multi-centre support for interdisciplinary cooperation and an easy-to-use API to facilitate integration with existing software. Furthermore, an extensible plugin system for easy adaptation to specific use cases should be included and an image-set administration aspect to manage and group images with restricted access through a user management system. Bounding boxes and polygon annotations as well as single click support are critical features for an efficient and flexible annotation workflow. Annotation templates enforce a unique naming scheme essential for standardisation and allow the incorporation of background knowledge. Additionally, guided screening to annotate WSIs systematically should be supported. Finally, to achieve reproducible results in the machine learning algorithm development process, a version control system for annotations and the possibility to perform inference of deep learning models is advantageous. Based on these requirements, we introduce EXACT, a novel online open-source software solution for massive collaboration in the age of deep learning and big data. EXACT was developed with seamless interaction to offline clients in mind, and interoperates with the established SlideRunner software[1].

In the following section, we describe the architecture of EXACT with its key features (see Table 1) and the design principles behind them. In the chapter "EXACT's applications", we showcase four very different projects where EXACT was applied to create high-quantity and high-quality data sets. Finally, we present a discussion and outlook.

EXACT's architectural design and features

The development of EXACT was based on the established online open-source software ImageTagger[8], which was developed for the RoboCup competition to create training data for machine learning projects. It already fulfils many of our basic requirements. Due to its low complexity, it allows for fundamental changes to the software design which are necessary to integrate functions like image set versioning. ImageTagger uses Django as its web framework, a Postgres database system and hypertext markup language (HTML) with JavaScript as frontend user interface. The following basic features and modules are substantially extended from or added to ImageTagger: We have added the Docker encapsulation, implemented the complete REST-API and have changed the image viewer to support the open-source software OpenSeadragon, which provides functionality to view WSIs in the browser. In this context, we have extended the images module to handle WSIs and provide functions to convert images into compatible WSI formats. Furthermore, we have made many performance adjustments to transfer annotations in parallel, display multiple annotation types simultaneously and synchronise annotations of other users. Also, we have completely redesigned the image viewer to display thumbnails of the image set and have created the possibility to include plugins. In the following subsections, we will first describe the architecture including the application and presentation tier. This is continued by introducing additional aspects of this software and their specialised extensions, like inference, data privacy, annotation maps, image set versioning, crowd-sourcing and annotation templates. Further implementation details are provided via videos, Jupyter notebooks or setup and code files in the "Supplementary information" section (Table 2).

Section: Architecture	
docker-compose.prod.yml	Setup EXACT via Docker with the command
	docker-compose -f docker-compose.prod.yml up -d –build
Supplementary Video S10	EXACT installation guide with Docker
Section: Application tier	
models.py	Saves information to the database or file system
views.py	Creates HTML views for the presentation tier
serializers.py	Serialises data for the REST-API
api_views.py	Provides database CRUD operations via the REST-API
doc/train_object_detection.ipynb	Code to train an object detection model via the REST-API
urls.py	Handles the mapping between URLs and python functions
exact/images	The *images* module is responsible for all image-based CRUD operations
Supplementary Video S7	How to create image sets and upload images
Supplementary Video S8	Explain image set details
exact/annotation	The *annotation* module is responsible for all CRUD operations regarding annotations, verification, media files and the annotation versioning system
exact/plugin	The *plugin* module handles analysis or visualisation plugins
exact/users	The *users* module handles the CRUD operations for users and teams
Supplementary Video S14	How to setup user access rights
exact/datasets	The *datasets* module provides features to automatically download and setup predefined data sets with their annotations
exact/datasets/templates	Folder containing data set HTML templates
exact/datasets/views.py	Implements functions to setup predefined data sets within EXACT
Supplementary Video S5	How to setup and use a demo data set
Section: Inference	
doc/Inference Asthma.ipynb	A REST-API inference example
Supplementary Video S9	Example for REST-API and JavaScript inference
Section: Annotation map screening mode	
doc/AnnotationMap.ipynb	Code to create annotation maps
Supplementary Video S2	How to create annotation maps
doc/ClusterCells.ipynb	Code to cluster Asthma cells
Supplementary Video S4	How to cluster Asthma cells
Section: Image set versioning and machine learning support	
doc/DownloadStudyAnnotations.ipynb	Code to download annotations from EXACT
Supplementary Video S15	How to create a new image set version and track changes
Section: Annotation templates	
Supplementary Video S3	How to create annotation templates with EXACT
Section: Pathology annotation study	
Supplementary Video S1	How to download annotations and explanation of parts of the annotation study
doc/DownloadStudyAnnotations.ipynb	Code to download annotations via the REST-API
Section: Multi-species pulmonary hemosiderophages cytology data set	
Supplementary Video S6	How to create density maps and explanation of the EIPH plugin
doc/Create_DensityWSI-Equine.ipynb	Source code for density maps
Section: Skin tumour tissue quantification	
doc/SyncImageAndAnnotations.ipynb	Code to synchronise between SlideRunner and EXACT
Supplementary Video S12	How to synchronise between SlideRunner and EXACT
Supplementary Video S11	How to segment with EXACT
doc/Segmentation.ipynb	Code to download information from EXACT to train a segmentation network
doc/PatchClassifier.ipynb	Code to download information from EXACT to train a patch classifier
Section: Clustering and visualisation of killer whale sounds	
Supplementary Video S13	How to perform sound clustering and visualisation

Table 2. EXACT documentation and references for the corresponding sections and supplementary videos.
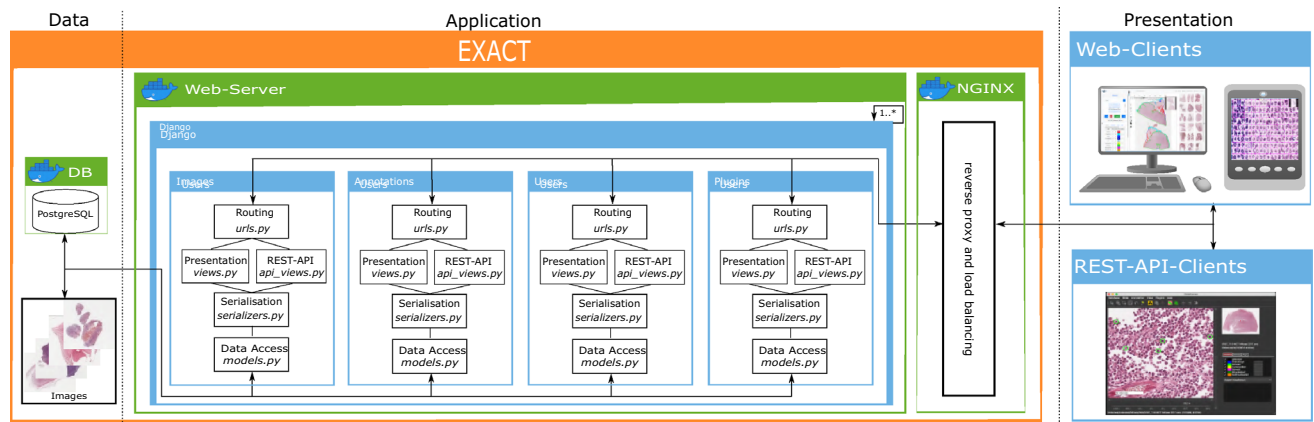
Figure 1. EXACT's three-tier architecture. Left: The data tier contains the PostgresSQL Docker container and the images, which can be saved within the Docker container or on the file system. Center: The application tier with the web-server Docker container instantiating Django instances with the corresponding modules. These modules handle images, annotations, users and plugin requests from the presentation tier and access the data tier to retrieve the stored information. NGINX works as a reverse proxy and handles EXACT's load balancing. Right: The presentation tier contains the EXACT web client or third-party applications like SlideRunner, which send requests via the provided REST-API.

Architecture. EXACT supports Docker to facilitate deployment and to enable a wide range of installation scenarios ranging from single-user, single-computer setups to massive cloud deployment with modern load balancing mechanisms (see Supplementary Video S10). EXACT is designed as a three-tier architecture containing the data, application, and presentation tier (Fig. 1). While the data and application tier are capsuled within Docker containers, the presentation tier is executed at the client side in HTML and JavaScript. This tier-based approach supports the development of secure applications by enforcing clearly defined interfaces between tiers and ensures that data access pipelines can not bypass tiers. The data tier includes a Postgres database system and the uploaded images and provides its content exclusively to the application tier.

Application tier. The application tier accesses the data tier to save information and to provide it to the presentation tier via a REST-API or as rendered HTML pages. EXACT uses Django as its web framework with four main modules (see Fig. 1, namely the images, users, annotations, and plugins modules). Each module is responsible for one group of tasks and is as independent as possible from the other modules. All modules implement functions for saving information to the database or file system and for creating HTML views. Furthermore, the modules define how to serialise data and provide a REST-API and a route request.

The images module is responsible for all image-based create, read, update and delete (CRUD) operations, and provides the logic to save all supported image formats and to provide them as a complete image or in a tile-based manner for WSIs. This multi type image support is implemented by converting all uploaded images that are not compatible with OpenSlide[9] into an OpenSlide compatible format, if supported, and saving them as an image pyramid. The formats and scanners that are supported by OpenSlide or our converter pipeline is listed in table 1. EXACT's open-source codebase allows developers to extend the list of supported image formats to their requirements and image dimensions. An example for multi-dimensional image data support is the audio video interleave (.avi) format. To support videos, EXACT converts each frame and handles the set of images as a individual WSIs with OpenSlide (Fig. 3). Additionally, the images module contains the image sets functionality which basically act as folders for the images and are assigned to teams to monitor user access rights. The Supplementary Videos (S7, S8) describe the creation of image sets and the upload of images.

The annotation module is responsible for all CRUD operations regarding annotations, verification, media files and the annotation versioning system. The annotation model saves annotation information about the annotation type, image, the creator and last editor with time stamps, JSON based meta data and the vector of coordinates to the database. The vector information is saved as JSON and contains the image coordinates of the annotation. The advantage of using JSON to store coordinates is the ability to search for annotations using vector coordinates in SQL. Furthermore, JSON provides the flexibility to adapt the representation of the vector to the target image format and dimensions.

The plugin module handles analysis or visualisation plugins which are specialised for specific research questions or data sets. One of these plugins is a persistent user-based screening mode which enables the user to systematically screen a WSI or parts of it on a self-defined zoom level (Fig. 2). This plugin, which is crucial to create high quality data sets, is implemented and used in the following manner: the user defines a zoom level and the algorithm divides the WSI in equal-sized patches with an overlap of 15% and saves the calculated screening map to the database. While the user is screening the WSI, the progress is constantly visualised at a thumbnail view of the WSI and the user's position on the WSI is saved to the database to recover the position if the screening has to be continued later.
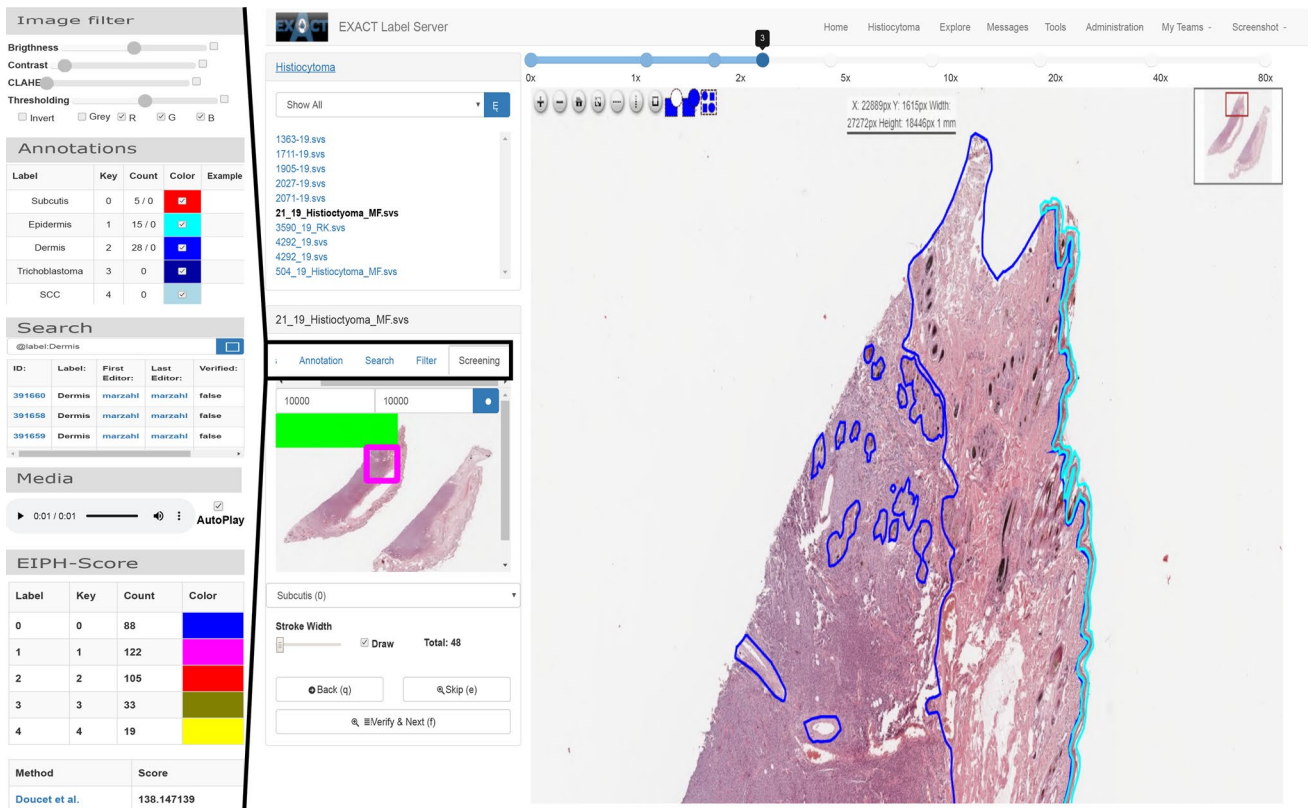
Figure 2. Left: Five examples of plugins (from top to bottom): The image filter plugin allows to make common intensity adjustments to the image, the annotation plugin shows the available annotations and their frequency of use. The search field allows to query the database for arbitrary annotation properties. The media plugin can be used to play media files attached to an annotation. The EIPH-Score plugin is an example of a domain-specific plugin, allowing to calculate the Doucet score[10]. Right: A screenshot of the annotation view depicting a WSI with polygon annotations, the list of images in the image set and the screening mode plugin, which enables the user to screen the image persistently. The screening plugin visualise the screened area in green and a purple rectangle for the current field of view.
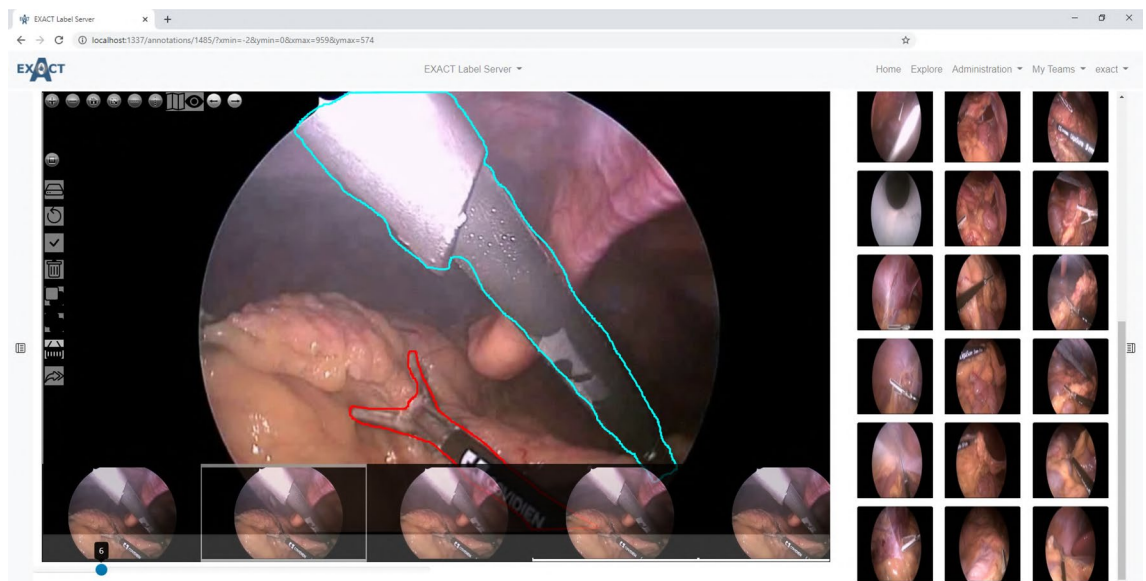
Figure 3. Left: An example frame from a laparoscopic colorectal video with annotated surgical instruments. Below the current frame (6), small previews of the previous and the next frames are displayed. Right: A browsing view to provide an overview of different image sets from the Robust Endoscopic Vision Challenge 2019[11].

The users module handles the CRUD operations for users and teams, and it further manages the user access rights. It is therefore involved in every server request to check if the request has the necessary CRUD rights (see Supplementary Video S14). To keep the annotations consistent, deleted users are anonymised and deactivated while their annotations are left unchanged.

An additional module is the data sets module. It provides features to automatically download and setup predefined data sets with their annotations from the EXACT user interface. The list of available data sets can be extended by adding an HTML template, which provides background information like the number of images or the data set source, and by implementing a download and setup function (see Supplementary Video S5).

Presentation tier. The presentation tier is programmed in HTML and JavaScript. All dynamic web-page contents like annotations, images or sub-images (tiles) for WSIs are loaded via JavaScript over the REST-API. The pagination-based REST-API implementation allow to load information chunk-wise from the server and therefore enable the transfer of huge quantities of data (e.g., hundreds of thousands of annotations per WSI) in parallel. We incorporated the open-source software OpenSeadragon as JavaScript-based image viewer with WSI support. A visualisation of the presentation tier is shown in Fig. 2.

Inference. Different modes for inference of deep learning models are supported to match the requirements across different use cases. In general, the inference can be performed directly on the server. For applications that require fast response times, the execution of JavaScript-based TensorFlow models is implemented by initially transferring the deep-learning model for the corresponding modality from the server via the REST-API to the JavaScript client. Afterwards, the model is executed on the current field of view of the image. The resulting annotations can then be rejected or confirmed and transferred to the server. For high-throughput applications, the inference load can be distributed on multiple machines by downloading the model and the WSIs via the REST-API and synchronising the results after performing inference. An inference example for equine asthma cytology images can be accessed at doc/Inference Asthma.ipynb or as Supplementary Video S9.

Data privacy and multi-centre support. Medical data should naturally be subject to the highest safety standards possible. Despite that, in order to enable interdisciplinary medical research and cooperation between different groups and locations, it can be necessary to share medical image data anonymously and in strict consideration of data privacy. Therefore, EXACT ensures the original image data, which may contain patient information (file name, metadata) to remain within the original institution while the actual data exchange between experts and institutes is executed on small sub-images via decentralised image storage. Technically this was implemented in several steps. Firstly, all server communication is protected with Hypertext Transfer Protocol Secure (HTTPS) and access is restricted via a user authentication system. Secondly, when transferring the images to an EXACT server instance, a new private name derived from the file name and a pseudonymised public name is generated. The pseudonymised public name is generated by the current date-time followed by a four-digit hash function of the original name (yymmdd-hhmm-****). Thirdly, for cooperation between different institutes, virtual image sets are supported. Here the information (for example annotations) is imported from several EXACT instances to a central server. However, access to the images themselves is always provided by the institute owning the data in compliance with their respective data privacy policy for images. This means that only the requested raw pixel data for the field of view is transferred to the collaborator, but not the image container or any metadata.

Annotation map screening mode. For applications that focus on annotation quality[1,7,8], a specialised validation mode is implemented that allows for a verification of each individual annotation. For data sets with hundreds or thousands of annotations, this is an important but error-prone, labour-intensive and time-consuming task. This becomes even more complicated for usage scenarios where each cell can receive multiple labels by one or multiple users. To make this validation process more convenient, we propose so-called annotation maps which can be efficiently processed using the screening mode. Annotation maps visualise all annotations belonging to one label in a matrix-like fashion which makes it easy to identify outliers. For efficient handling, a new image is created for each class which consists of all corresponding annotations which can then be viewed in the screening mode (Fig. 4 top and Supplementary Video S2). The annotation maps can be efficiently screened for errors, while the users can define how many annotations they want to see simultaneously. Corrections made on these screening images are synchronised with the original data.

An advanced extension of this method is the clustering of labelled and unlabelled images or image patches. This manner of presentation allows the user to efficiently create initial labels or to quickly validate prior annotations, since similar images which are likely to have similar labels are displayed closely together. The clustering pipeline consists of three steps. Firstly, characteristic features are extracted from each image, for example, by deep learning or classic image processing. Secondly, the extracted high-dimensional features are transformed into two-dimensional features, for example, using t-SNE[12], PCA[13] or UMAP[14]. Finally, the extracted image patches are drawn in a new image container according to their nearest two-dimensional feature representation, which does not overlay any other image patches. The resulting image is visualised for labelling or validation (Fig. 4 bottom) in EXACT. A detailed code example can be accessed at doc/ClusterCells.ipynb in combination with a Supplementary Video S4.

Image set versioning and machine learning support. In general, two main criteria in research and medical applications are reproducibility and traceability of results and experiments. Especially reproducibility is non-trivial in settings where researchers from different fields like medicine and computer science work together
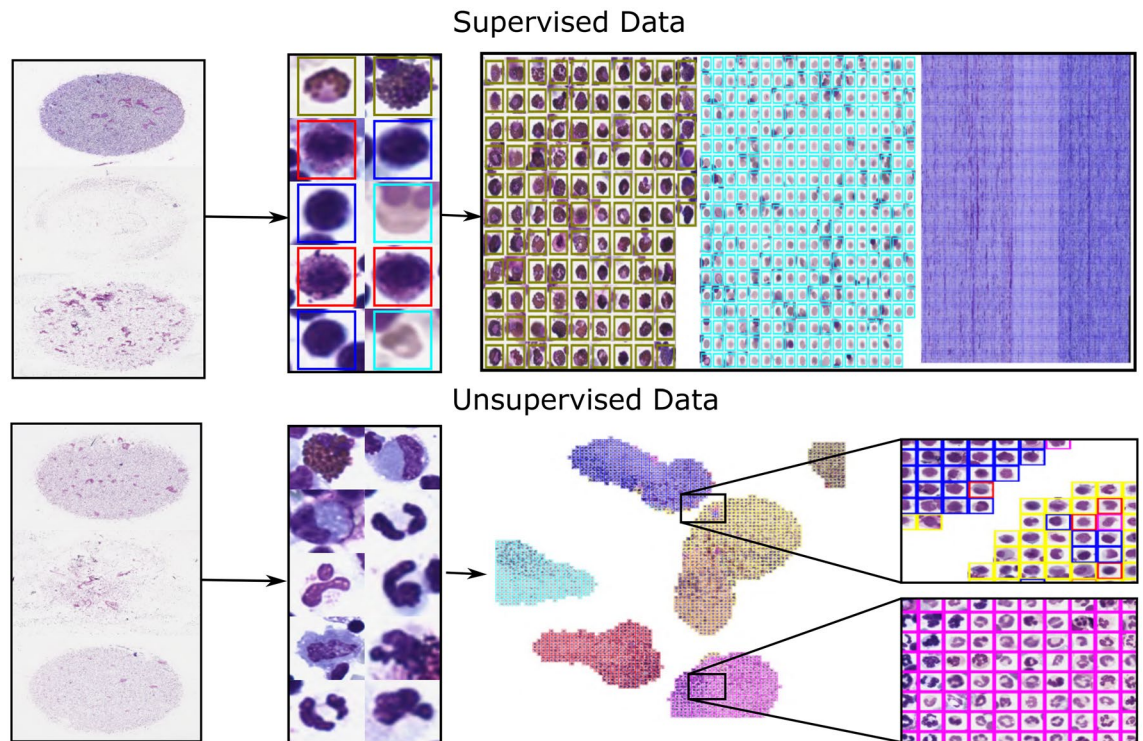
Figure 4. Top row: Supervised single-cell validation, with annotation maps generated from three labelled equine asthma WSIs where each colour represents one class of cells. Bottom row: UMAP[14] dimensionality reduction approach in an unsupervised setting. The segmented equine asthma cells are first classified. Then, features for each cell are extracted. Afterwards, the high-dimensional features are transformed into a two-dimensional representation and visualised in a new image. Both approaches allow the user to verify and enhance the automatic classification results.

and make adjustments to data sets over time. In software development, it is an established process to use version control systems (such as git or subversion) for source code to coordinate the collaboration between software developers and keep code changes traceable. Remarkably, this process is to our knowledge not provided by any open source software for annotations on medical data sets. To implement this feature, we included a versioning system with functions that support traceability of annotations and attach experimental results to versions. If a version is added to a data set, the current annotation state, an optional description, and the current list of images in the data set is saved. If a user leaves a project, he or she is not deleted from EXACT but only deactivated and anonymised so that versioned annotations are not affected. In contrast, if an image is deleted from the image set, all annotations are lost due to the impracticability of versioning WSIs with multiple gigabytes of size. For example for training machine learning algorithms, the annotations can be filtered by versions and exported in user-defined text formats or per script using the provided REST-API. This supports the users to perform experiments on defined, reproducible data sets while providing the flexibility to export input data to a wide range of machine learning frameworks. Additionally, training artefacts like performance metrics, annotations, or generated models can be uploaded and attached to a version. In combination with the virtual image set function introduced previously in this article, it is possible to create virtual training, testing, and validation sets. This combination of versions and virtual image sets helps to keep track of different experiment versions and supports the comparability of results (see Supplementary Video S15).

Crowd-sourcing and study support. One of the biggest challenges in developing, training, testing, and validating state-of-the-art machine learning algorithms is the availability of high-quality, high-quantity labelled image databases. Crowd-sourcing has numerous successful applications in the medical field[15] and crowd-algorithm collaboration has the potential to decrease the human effort[16]. EXACT supports this development by providing multiple features for managing crowd-sourcing. Firstly, the user privilege system allows to set specific rights like annotation or validation to users or user groups. Secondly, the crowd- or expert-algorithm collaboration is assisted by importing pre-computed annotations or generating them on-premise with machine learning models. Finally, EXACT supports multiple annotation modes like.

1. *Cooperative* One user can verify the image, and each user sees all other annotations.
2. *Competitive or blind* Every user must verify every image and cannot see other users' annotations.
3. *Second opinion* A predefined number of the users must verify every annotation.

Annotation templates.   Standardisation is critical to encourage cooperation, interoperability and efficiency. To support this, EXACT introduces annotation templates, which allow to define a set of properties of annotations associated with a defined label. Annotation templates contain general information about the target structure like a name, an example image, the sort order in which the annotation should be displayed on the user interface, display colour, keyboard shortcuts to efficiently assign the label to an annotation, and default size. Default sizes enable the user to introduce background knowledge into the annotation process; this allows for efficient single click annotations and reduces the need to further adjust annotations. One or more annotation templates are grouped to products with pieces of information like name or description and can be assigned to image sets. The products in turn can be assigned to multiple image sets and support the reproducibility of the annotation process by enforcing a standard naming and annotation schema (see Supplementary Video S3).

EXACT's applications
In the following sections, we present several previously published usage scenarios using EXACT and describe how they made use of EXACT's features to increase efficiency and annotation quality.

Pathology annotation study.   In a study by Marzahl et al.[17], EXACT was used to investigate how the efficiency of the pathology image annotation process can be increased with computer-generated pre-computed annotations. The design and results of the published study[17] showcase a prominent EXACT use case and are summarised in the following paragraphs. Ten pathologists had to perform three pathologically relevant diagnostic tasks on 20 images each, once without algorithmic support and once with algorithmic support in the form of pre-computed annotations which are visualised for the expert to review. Firstly, they had to detect mitotic figures on microscopy images. Each of the 20 images spanned ten high power fields (HPF, total area = 2.37 mm^2). The second task focused on performing a differential cell count in cytology of equine pulmonary fluid; a task relevant for diagnosing respiratory disease. For this, five types of visually distinguishable cells (eosinophils, mast cell, neutrophils, macrophages, lymphocytes) had to be labelled. The last task was to determine the severity of pulmonary haemorrhaging by grading the amount of breakdown products of red blood cells (hemosiderin) in alveolar macrophages according to the scoring scheme by Golde et al.[18].

Several EXACT features were used for this study. First of all, we used the blind annotation mode for assigning identical grading tasks to all pathology experts, which we then combined with the feature of importing pre-computed annotation for the algorithmic support. The annotation templates enabled rapid single click annotations by providing appropriate default annotation sizes for each cell type, which was particularly helpful for the equine asthma task where the different cells types have notable size differences. The systematical grading of the images was supported by the persistent screening mode plugin, which enables the expert to resume the grading process at the previously selected position on the slide at any time. During the course of the study, the pathologists annotated 26,015 cells on 1200 images. The algorithmic support with EXACT lead to an increase in accuracy and a decrease of annotation time[17] for all tasks. For detailed results, we kindly refer the reader to the original study. A video showcasing this study can be viewed (see Supplementary Video S1) with related source code at doc/DownloadStudyAnnotations.ipynb to download the annotations. Furthermore, we added the images and ground-truth annotations from the study to the list of demo data sets which can be accessed and instantiated from the EXACT user interface.

Multi-species pulmonary hemosiderophages cytology data set.   In our previous work[19], 17 WSIs with 78,047 pulmonary hemosiderophages were fully annotated by a veterinary pathologist and used to develop a deep learning based object detection model. Pulmonary haemorrhage is diagnosed by performing a cytology of bronchoalveolar lavage fluid (BALF). The basis for this scoring system from Golde et al.[18] is that alveolar macrophages degrade the red blood cells into an iron-storage complex called hemosiderin. After staining the sample with Perls' Prussian Blue or Turnbull's Blue, the macrophages can be assigned a discrete grade from zero (low hemosiderin content) to four (high hemosiderin content).

Building on this work, EXACT played an essential part in creating a large, fully annotated multi-species pulmonary haemorrhage data set. For this project 40 additional equine WSIs, seven feline WSIs and twelve human WSIs with evidence of chronic pulmonary haemorrhaging were annotated by expert-algorithm collaboration using EXACT and the provided object detection model. In a first step, all WSIs were annotated automatically with the deep learning model and afterwards a pathologist carefully reviewed whether all target objects were annotated. Then, the pre-computed label class was verified separately by incorporating and modifying EXACT's novel annotation map feature based on a cell-based regression approach[19] that reflects the continuous increase of the hemosiderin content in the target cells. This approach assigns a continuous grade between zero and four to each cell to create the annotation map for efficient manual validation (Fig. 5 bottom). This annotation map orders the cells by score on the x-axis resulting in a density map of hemosiderin scores. By stacking the corresponding cell images of the same score along the y-axis, the quantity of annotated cells across the different scores is visualised (Fig. 5 bottom). This enables the trained pathologist to efficiently verify the computer-generated label class by focusing on the cells which are located on the borders between two grades. Another specialised plugin was developed to calculate the EIPH score over the current field of view in real-time (Fig. 2), according to Doucet et al.[10]. Code to create density maps can be accessed at doc/Create_DensityWSI-Equine.ipynb in combination with a Supplementary Video S6.

Skin tumour tissue quantification.   This ongoing project aims to segment and classify nine of the most common dog skin tumour types with deep learning algorithms. For this purpose, slides were scanned and partly annotated using SlideRunner's advanced tissue annotation tools. This project needs to synchronise the generated
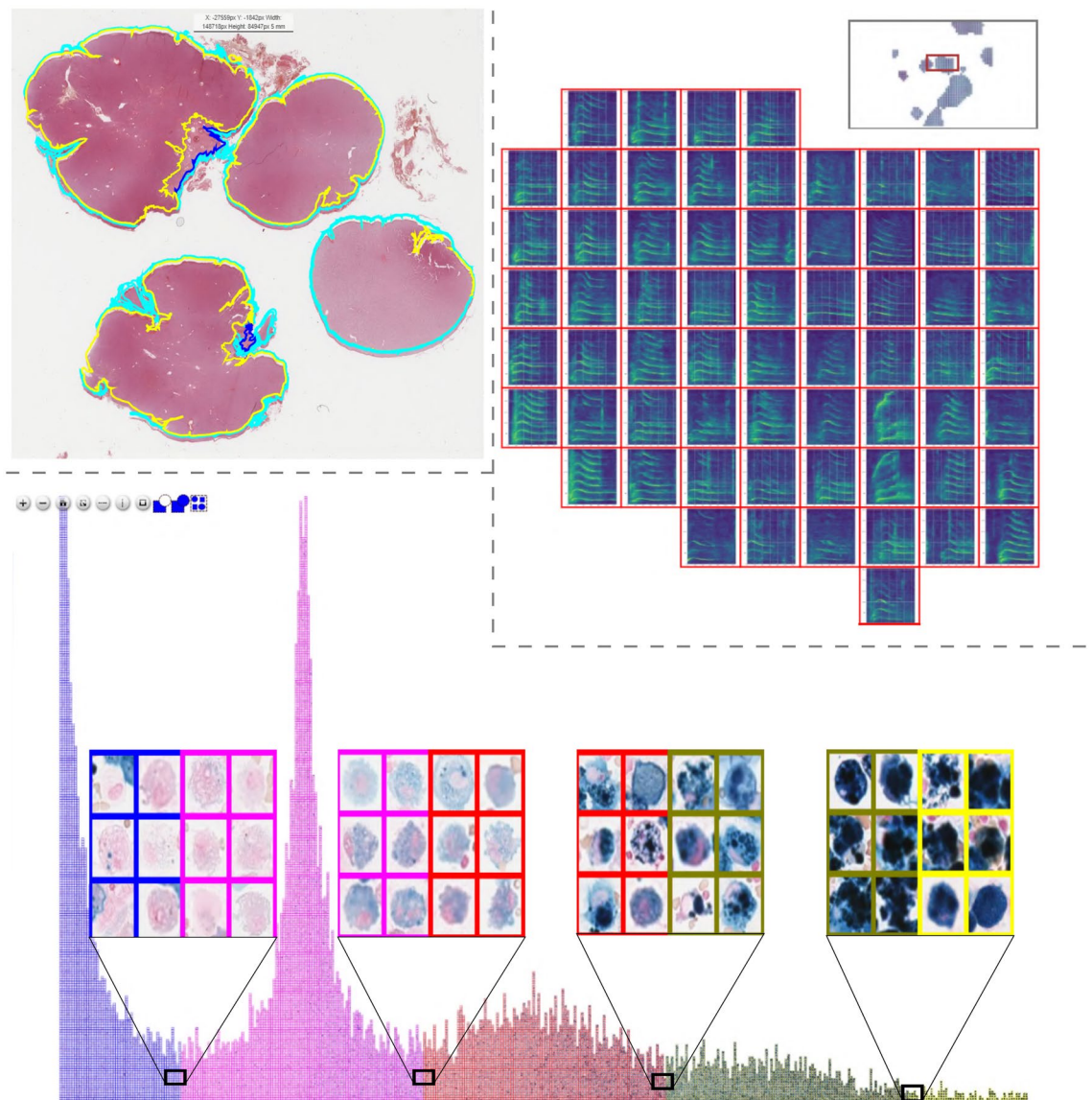
Figure 5. Top Left: Polygon annotations of a canine skin tumour tissue whole slide image. Top Right: Clustered whale sound spectroscopy images with the option to listen to the attached waveform online. Bottom: Pulmonary hemosiderophages, labelled according to their predicted class and arranged according to their predicted regression score for efficient validation by human experts.

slides and annotations to EXACT for coordination and distribution between the participating pathology experts and computer scientists for analysis at multiple institutes and locations. SlideRunner and EXACT communicate via EXACT's REST-API to synchronise annotations, images and annotation templates (see Supplementary Video S12). EXACT's novel feature of annotation templates plays an essential role in increasing standardisation and the overall image set quality by ensuring standard annotation naming schemes and the use of polygon annotations independent of the user or user application (Fig. 5 top left). While the project is actively being developed, 350 slides have already been fully annotated, resulting in 12,859 polygon annotations representing tissue layers. This indicates that a combination of online and offline tools enables fast multi-expert annotations. Code to download images, annotations and train a segmentation model can be found at doc/Segmentation.ipynb in combination with a Supplementary Video S11.

Clustering and visualisation of killer whale sounds.   While the EXACT platform is primarily developed for cooperative interdisciplinary research on microscopy images, its flexibility extends to other research areas without adaptation. We therefore showcase its use in a project that aims at deepening the understanding of killer whales *(Orcinus Orca)* and their large variety of different sound types[20]. In this study, EXACT is used to cluster and visualise the spectral shape of machine-pre-segmented killer whale audio samples (Fig. 5 top right). Multiple EXACT features support this challenging undertaking: firstly, the support of viewing and annotating gigapixel size images, which, in this use case, contain up to thousands of clustered spectrograms, where each

spectrogram represents an individual killer whale sound. Secondly, grouped annotation assignments, which enable the user to select numerous visually grouped spectrograms simultaneously by drawing a rectangle around them in order to assign them to the same label. Finally, EXACT supports attaching media records like videos, images or sound files to the respective annotations and plays them in a web browser (Fig. 2 left). These features enable the user to see the grouped spectrograms and additionally listen to the attached killer whale sound (see Supplementary Video S13).

Discussion

With the rapidly evolving digitisation of image data and the widespread use of machine learning algorithms, the need for platforms that are able to organise and display large amounts of large image data while also managing and keeping track of annotations is more crucial than ever. In this paper, we have introduced EXACT which is an open-source online platform that enables the collaborative interdisciplinary analysis of images with annotation version control.

EXACT has proven to satisfy these requirements in several different projects ranging from collaborative tissue segmentation in the field of digital pathology to whale sound clustering. This diverse range of application represents its primary advantage. It does not only allow to extend existing offline projects with cooperation and synchronisation functions, but is also able to support researchers in various fields. Furthermore, EXACT's features provide computer scientist with version controlled annotations, advanced visualisation techniques like annotation maps or clustering, and saving artefacts from experiments like trained models. With EXACT, it is also possible to define reproducible training, validation and testing sets. Generally, all software solutions face the issues of support, maintenance and handling future developments. To increase the chances of turning EXACT into a successful project which offers added value for the community in the long term, EXACT will stay open-source and focus on the compatibility and synchronisation with other image analysis software. The flexible open-source software architecture allows for adaptation to future developments in digital pathology or other research areas. In future releases, we are planning to support a higher amount of publicly available data sets. In addition, we want to create specialised plugins exploring molecular pathology issues—an increasingly significant subdiscipline of the classic anatomical pathology. Also, valuable future extensions to EXACT include the integration of servers (like Omero[21]), which are specialised in providing microscopic images, as well as exploring options to connect EXACT with other established tools like Cytomine. Furthermore, we are investigating the integration of gamification as a promising new method to annotate data at scale.

In summary, EXACT provides a novel feature set to boost the creation of high-quality big data sets in combination with functions to develop state-of-the-art machine learning algorithms.

Code availability

References

1. Aubreville, M., Bertram, C., Klopfleisch, R. & Maier, A. Sliderunner. In *Bildverarbeitung für die Medizin 2018* (eds Maier, A., Deserno, T., Handels, H., Maier-Hein, K., Palm, C. & Tolxdorff, T.) 309–314 (Springer Vieweg, Berlin, Heidelberg, 2018).
2. Hollandi, R., Diósdi, Á., Hollandi, G., Moshkov, N. & Horvath, P. Annotatorj: An imagej plugin to ease hand-annotation of cellular compartments. *Mol. Biol. Cell* **31**(20), 2179–2186 (2020).
3. De Chaumont, F. *et al.* Icy: An open bioimage informatics platform for extended reproducible research. *Nat. Methods* **9**, 690 (2012).
4. Bankhead, P. *et al.* Qupath: Open source software for digital pathology image analysis. *Sci. Rep.* **7**, 1–7 (2017).
5. Marée, R. *et al.* Collaborative analysis of multi-gigapixel imaging data using cytomine. *Bioinformatics* **32**, 1395–1401 (2016).
6. Puttapirat, P. *et al.* Openhi—An open source framework for annotating histopathological image. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 1076–1082 https://doi.org/10.1109/BIBM.2018.8621393 (2018).
7. Obando, D. F. G., Mandache, D., Olivo-Marin, J.-C. & Meas-Yedid, V. Icytomine: A user-friendly tool for integrating workflows on whole slide images. In *Digital Pathology* (eds Reyes-Aldasoro, C. C., Janowczyk, A., Veta, M., Bankhead, P. & Sirinukunwattana, K.) 181–189 (Springer International Publishing, Cham, 2019).
8. Fiedler, N., Bestmann, M. & Hendrich, N. Imagetagger: An open source online platform for collaborative image labeling. In *Robo-Cup 2018: Robot World Cup XXII* (eds Holz, D., Genter, K., Saad, M. & von Stryk, O.) 162–169 (Springer International Publishing, Cham, 2018).
9. Goode, A., Gilbert, B., Harkes, J., Jukic, D. & Satyanarayanan, M. Openslide: A vendor-neutral software foundation for digital pathology. *J. Pathol. Inform.* **4**(1), 27 (2013).
10. Doucet, M. Y. & Viel, L. Alveolar macrophage graded hemosiderin score from bronchoalveolar lavage in horses with exercise-induced pulmonary hemorrhage and controls. *J. Vet. Intern. Med.* **16**, 281–286 (2002).
11. Maier-Hein, L. *et al.* Heidelberg colorectal data set for surgical data science in the sensor operating room. arXiv:2005.03501 (2020).
12. Maaten, L. V. D. & Hinton, G. Visualizing data using t-sne. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
13. Wold, S., Esbensen, K. & Geladi, P. Principal component analysis. *Chemometr. Intell. Lab.* **2**, 37–52 (1987).
14. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. arXiv:1802.03426 (2018).
15. Ørting, S. N., Doyle, A., van Hilten, A., Hirth, M., Inel, O., Madan, C. R., Mavridis, P., Spiers, H. & Cheplygina, V. A survey of crowdsourcing in medical image analysis. *Hum. Comput.* **7**(1), 1–26 (2020).
16. Marzahl, C. *et al.* Is crowd-algorithm collaboration an advanced alternative to crowd-sourcing on cytology slides? In *Bildverarbeitung für die Medizin 2020* (eds Tolxdorff, T., Deserno, T., Handels, H., Maier, A., Maier-Hein, K. & Palm, C.) 26–31 (Springer Fachmedien Wiesbaden, Wiesbaden, 2020).

17. Marzahl, C. *et al.* Are fast labeling methods reliable? A case study of computer-aided expert annotations on microscopy slides. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020* (eds Martel, A. L., Abolmaesumi, P., Stoyanov, D., Mateus, D., Zuluaga, M. A., Zhou, S. K., Racoceanu, D. & Joskowicz, L.) 24–32 (Springer International Publishing, Cham, 2020).
18. Golde, D. W., Drew, W. L., Klein, H. Z. *et al.* Occult pulmonary haemorrhage in leukaemia. *Br. Med. J.* **2**, 166–168 (1975).
19. Marzahl, C. *et al.* Deep learning-based quantification of pulmonary hemosiderophages in cytology slides. *Sci. Rep.* **10**, 1–10 (2020).
20. Bergler, C. *et al.* ORCA-SPOT: An automatic killer whale sound detection toolkit using deep learning. *Sci. Rep.* https://doi.org/10.1038/s41598-019-47335-w. (2019).
21. Allan, C. *et al.* Omero: flexible, model-driven data management for experimental biology. *Nat. Methods* **9**, 245–253 (2012).

Acknowledgements

Author contributions

C.M. developed the server, created the visualisation code and wrote the main part of the manuscript. M.A. co-wrote the manuscript, provided code for the synchronisation with SlideRunner, provided expertise through intense discussions. C.A.B. co-wrote the manuscript, provided expertise through intense discussions. J.M., J.V., C.B., C.K., K.B., R.K., A.M. provided expertise through intense discussions. All authors contributed to the preparation of the manuscript and approved of the final manuscript for publication.

Funding

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41598-021-83827-4.

Correspondence and requests for materials should be addressed to C.M.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.