

Do hlubin digitální patologie

Jakub Kantner, Senior AI Researcher

MSD Czech Republic

October 14, 2026



About MSD

- A global, research-intensive biopharmaceutical company
- ~75000 employees, \$65B Revenue, \$15.8B R&D expenses (in 2025)
- Known as Merck in the USA and Canada, MSD globally
- Headquarters: Rahway, New Jersey, USA
- Prague: Tech Center - IT and Finance, ~1800 people

About me

Jakub Kantner, Senior AI Researcher

- Master's in Applied Mathematics, FNSPE CTU
- 9 years in MSD
- Started as student intern in DMV
- Responsible AI & AI Research
- Focus on
 - New models assessments
 - AI Safety
 - PoCs
- Hobbies
 - Hiking, coffee, drama club & improv

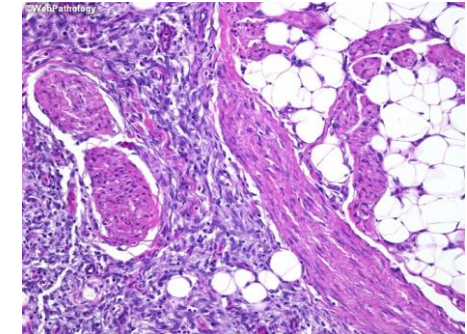


Digital Pathology Annotation Transfer Tool

Introduction

Pathology

- Pathology is the study of the causes, nature, and effects of disease.
 - Pathologist is medical doctor with additional training.
 - Histology is the study of tissues and cells under a microscope.



Introduction

Digital Pathology

Is a process by which histology slides are digitized to produce high-resolution images.

Quantitative image analysis

Enables scientists to gain insight into various pathological information in tissue samples.

Problem definition

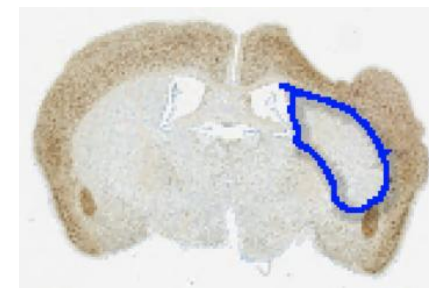
Target group

Pathologists performing annotation of large volumes of digital images

Manual tissue annotation

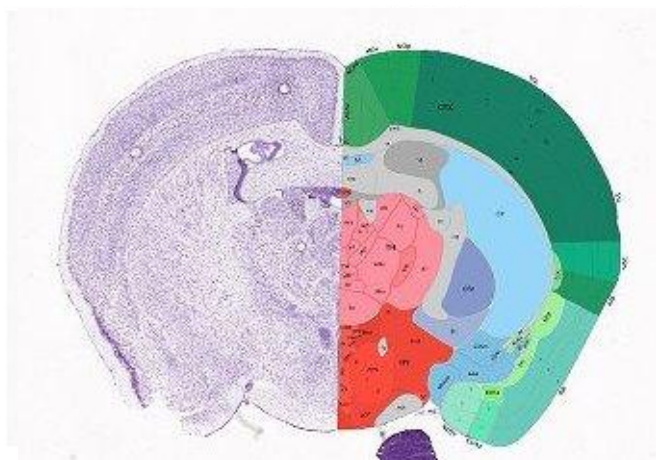
It is highly labor-intensive task and requires significant technical expertise.

It entails highly redundant and time-consuming work, which is not feasible for large studies (100+ slides)



Use case – Mouse brain annotation

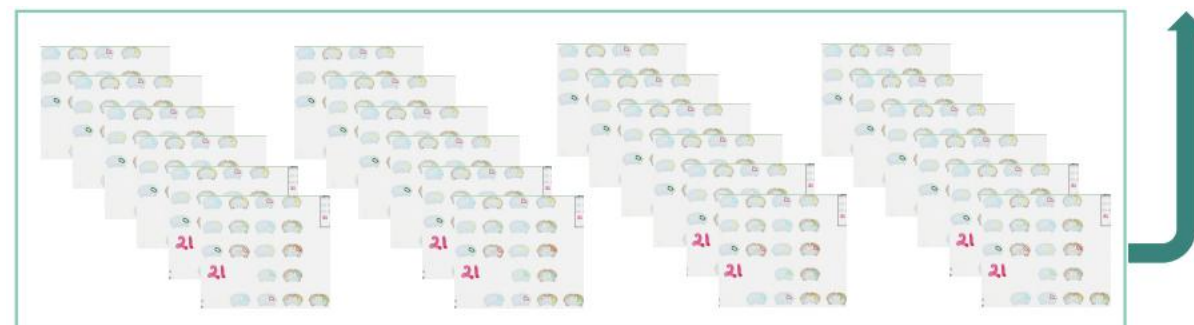
- Majority of all animals used for drug research studies are rodents.
- Transfer of annotations of standardized anatomical regions of interest to mouse brain.



Allen Mouse Brain Atlas

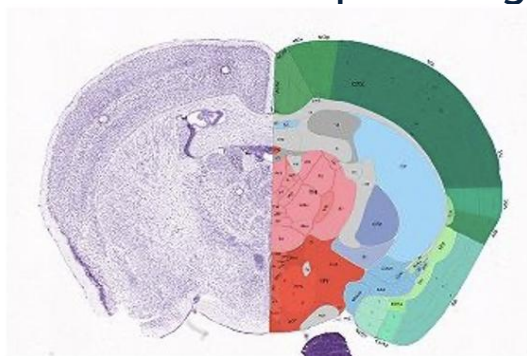
Example of single study

- 12-100 samples
- 6 slides per animal & stain
- 4 FTEs over 2 weeks

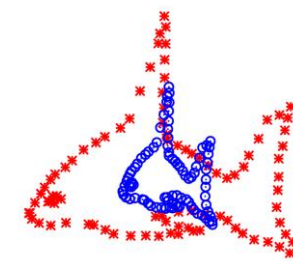
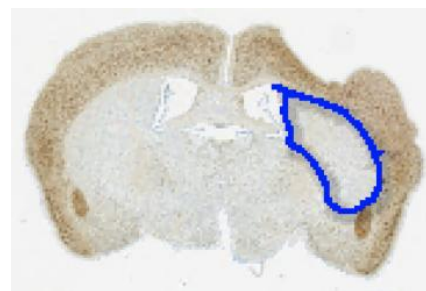


What have we developed?

- A software tool that has been developed to reduce the time required to annotate digital pathology slides in larger studies.



Allen Mouse Brain Atlas

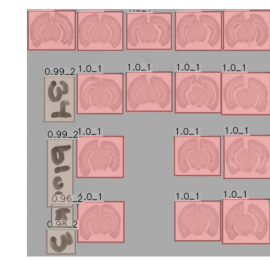


- **Image registration** finds optimal geometric alignment of morphologically similar and spatially corresponding digital pathology slides
- **Seamless integration** with existing infrastructure and business workflows

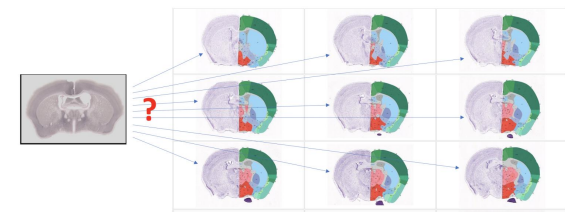
Pipeline overview

1. Individual brain detection
2. Automatic atlas layer selection
3. Automatic naming of annotations
4. Image registration

1



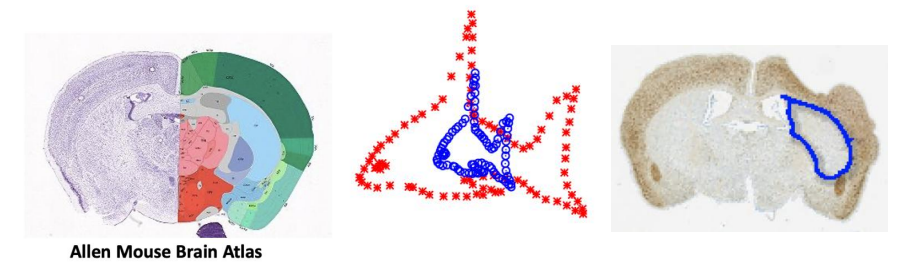
2



3

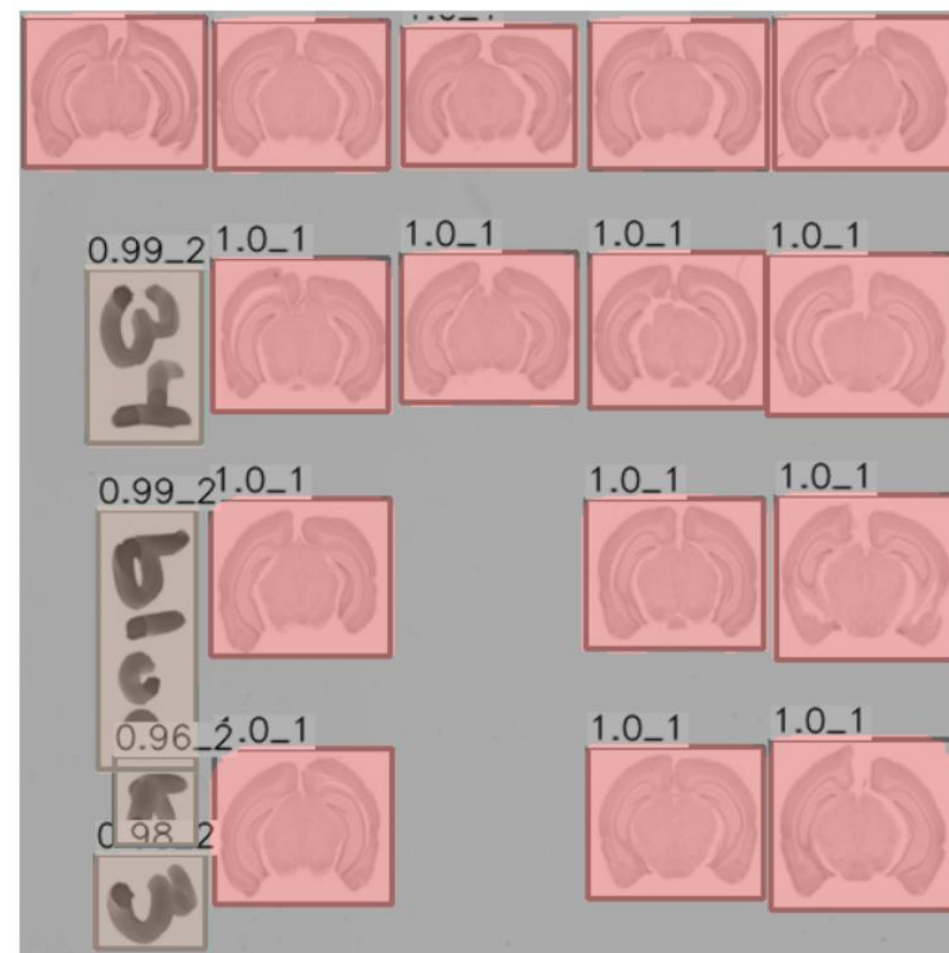
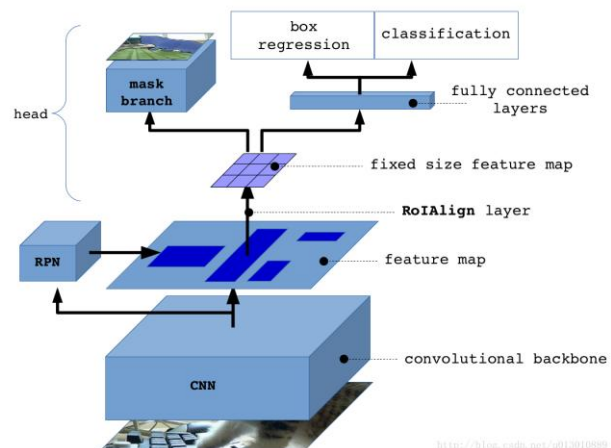
col/row	1	2	3	4	5
1	Pos. cntrl 1	Pos. cntrl 2	Pos. cntrl 6	Neg. cntrl 1	Neg. cntrl 5
2	Blank	Pos. cntrl 3	Pos. cntrl 7	Neg. cntrl 2	Neg. cntrl 6
3	Blank	Pos. cntrl 4	Blank	Neg. cntrl 3	Neg. cntrl 7
4	Blank	Pos. cntrl 5	Blank	Neg. cntrl 4	Neg. cntrl 8

4



Individual brain detection

- Mask-RCNN object detection model
- Pretrained model learned the task from 40 images
- Test - 98% of 400 brains correctly



What is Mask R-CNN?

An instance segmentation model that finds each object and outlines its pixels.

One prediction per object

Class + confidence	What is the object, and how confident is the model?
Bounding box	Where is the object in the image?
Instance mask	Which pixels belong to this particular object?

Detection vs. segmentation

Object detection gives each object a box.
Semantic segmentation labels pixels by class.
Instance segmentation also separates individual objects of the same class.

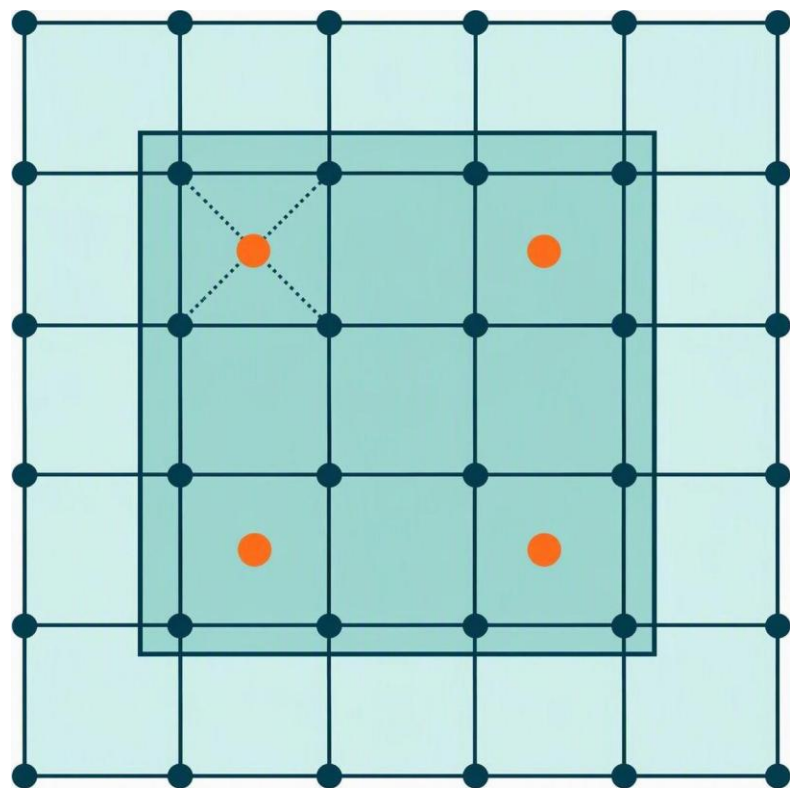
Example: two cars receive two separate masks, even though both have the class “car”.

How Mask R-CNN works

- 01 Feature extraction** A convolutional backbone converts the image into shared feature maps.
- 02 Region proposals** A Region Proposal Network (RPN) suggests boxes that may contain objects.
- 03 RoIAlign** Aligned sampling turns each proposed region into a fixed-size feature map.
- 04 Parallel predictions** One head predicts class and box coordinates. A second predicts an object mask.
- 05 Final instances** Score filtering and overlap suppression retain the detections and their masks.

RoIAlign preserves spatial detail

Accurate masks need features that stay aligned with the image.



The problem with rounding

RoI pooling rounds region coordinates to the feature grid.

Sampling between grid points

RoIAlign uses bilinear interpolation at fractional coordinates.

Weighted interpolation

$$f(x, y) = \sum w_i f_i, \text{ with } \sum w_i = 1$$

The four nearest values contribute.

Approach to fine-tuning Mask R-CNN

Training data

We annotated each object with its class, bounding box and mask.

We used separate training, validation and test images, keeping related images in the same split to avoid leakage.

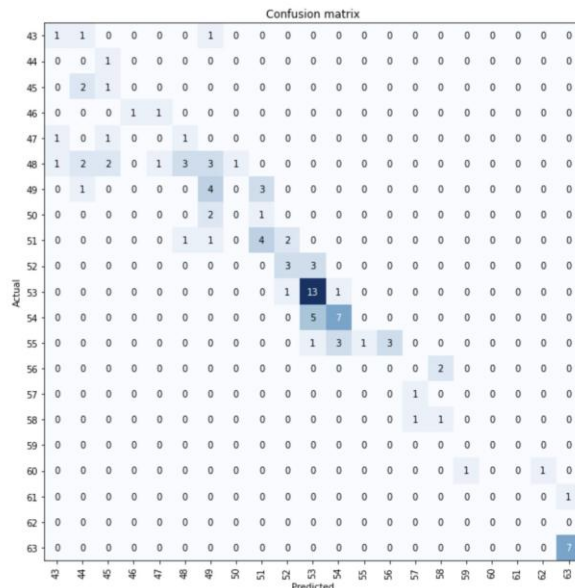
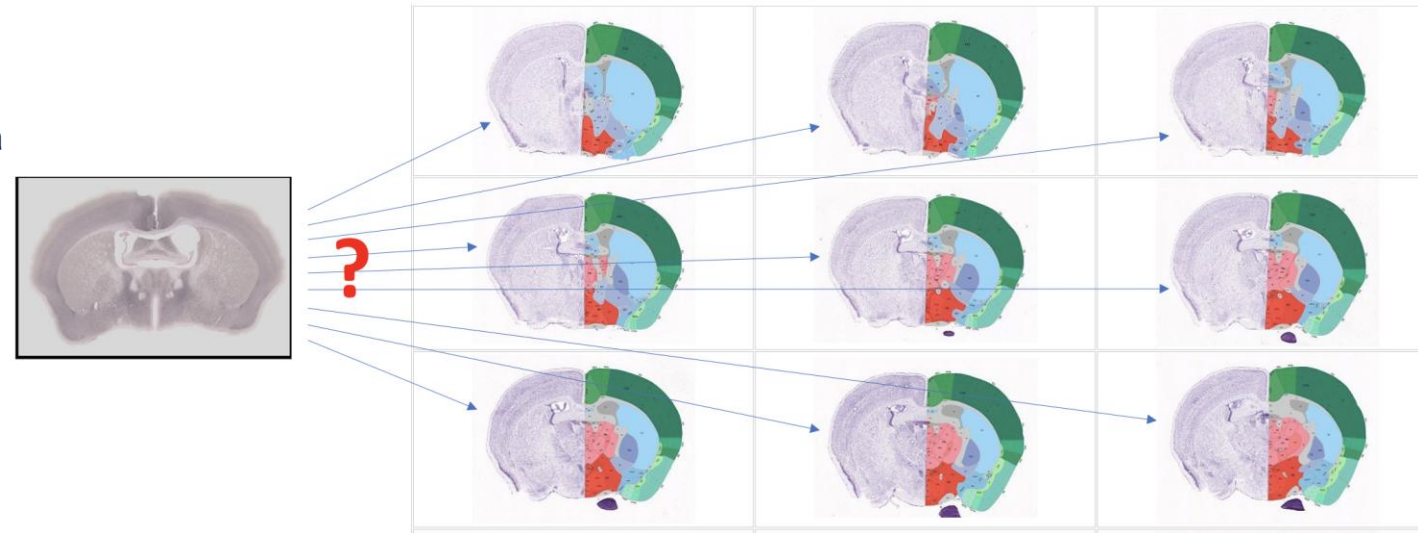
Training images updated the model weights.
Validation images guided our settings and model selection.

Adaptation of the model

We started with pretrained weights and replaced the class/box and mask prediction layers for our target classes.

We trained the new prediction layers while keeping the backbone frozen to retain its learned image features.

-



actual	45
predicted	44

actual	51
predicted	51

Allen Mouse Brain Atlas

An anatomical reference

reference maps of the adult mouse brain, alongside gene-expression data by the Allen Institute

Annotated sections define brain-region boundaries and a consistent hierarchy of region names.

The 2011 reference includes 132 coronal sections at 100 μm intervals and 21 sagittal sections*.

50 coronal sections used



EfficientNet-B0 for atlas matching

Our atlas-matching approach

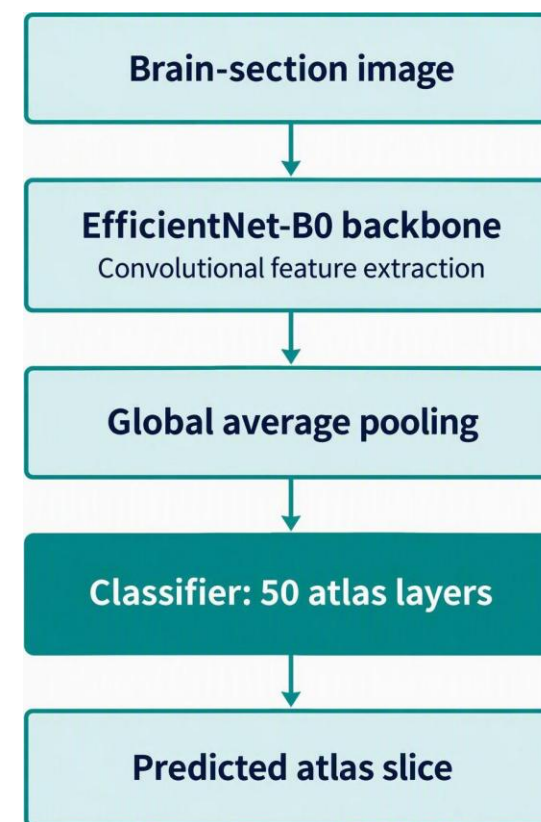
We used pretrained EfficientNet-B0 to classify each brain-section image by atlas layer.

We adapted the final classifier to our 50 atlas-layer classes, using 10 training samples per layer.

The model extracted visual features and assigned a score to each atlas layer. We used the highest-scoring class to select the reference slice.

Image registration then aligned the selected reference with the tissue image.

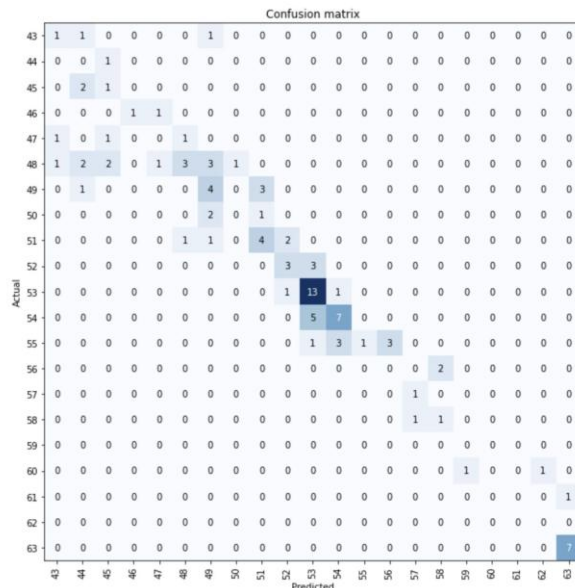
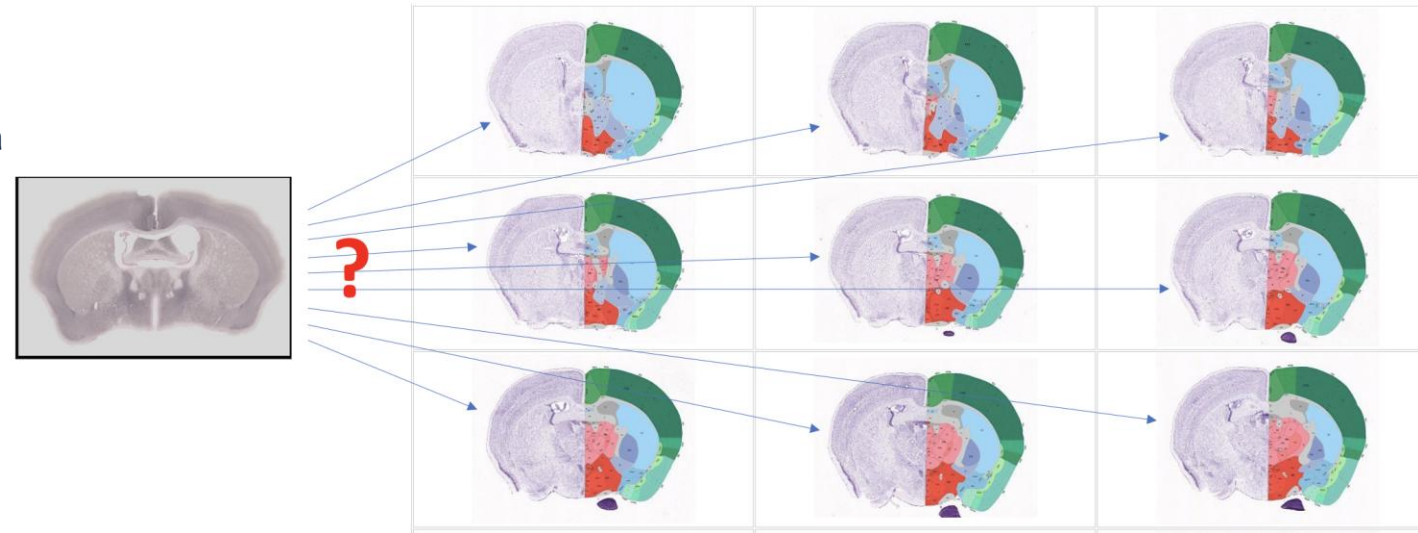
Simplified architecture



How a brain section becomes an atlas-layer prediction

- 01 Image preparation** A consistent crop, size and normalization prepare the brain-section image (extracted using Mask-RCNN)
- 02 Feature extraction** The pretrained convolutional backbone extracts visual patterns from the image.
- 03 Feature summary** Global average pooling summarizes each feature map into a single value.
- 04 Atlas-class scores** A task-specific classification layer produces one score for each of the 50 atlas layers.
- 05 Slice selection** Softmax converts scores to class probabilities. The highest score identifies the predicted slice.

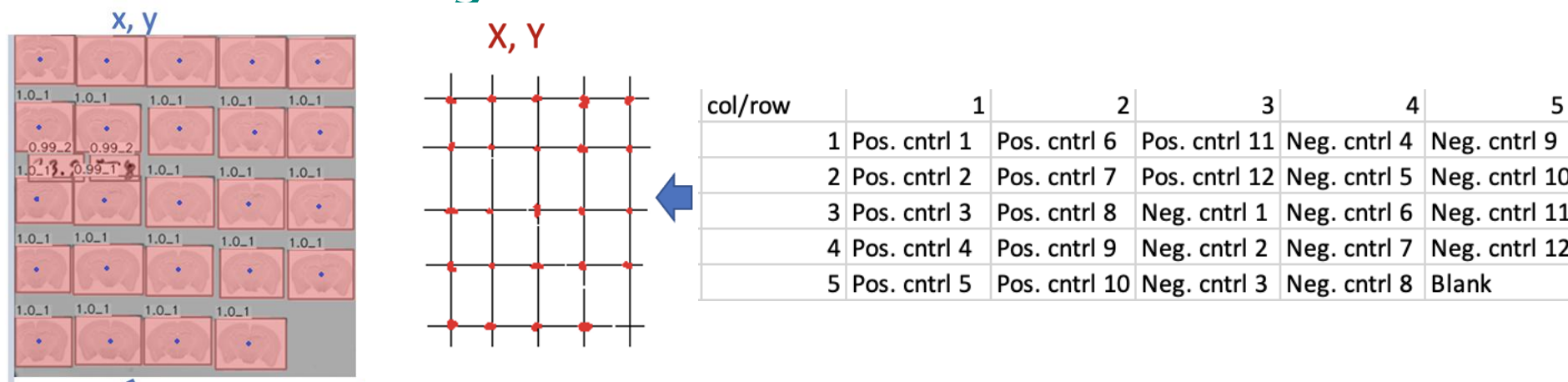
-



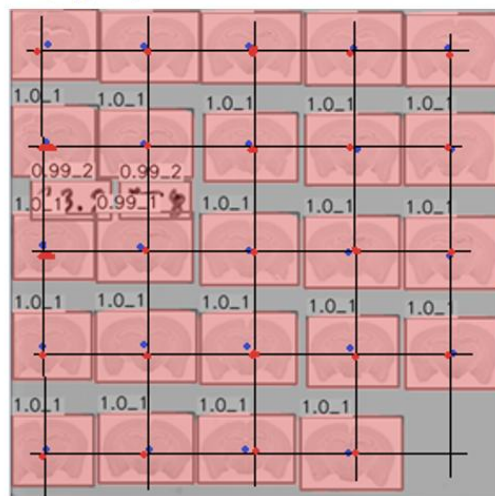
actual	45
predicted	44

actual	51
predicted	51

Automatic naming of annotations



x, y, X', Y'



- Annotation name based on predefined table
- Approach
 - Template to regular grid
 - Shift and scale grid to fit centers of bounding boxes
 - Measure distance between all grid points and all bounding box centers
 - Graph matching using Hungarian algorithm

Hungarian algorithm: theory

The linear assignment problem

Square case: n detections and n positions.

c_{ij} is the cost (distance) of detection i from position j .

minimize $\sum_i \sum_j c_{ij} x_{ij}$

$\sum_j x_{ij} = 1$ for every row i

$\sum_i x_{ij} = 1$ for every column j

$x_{ij} \in \{0, 1\}$

$x_{ij} = 1$ means “choose this pair.”

The constraints give each detection one position and use each position exactly once.

Why the reductions work

Subtracting a constant from a whole row or column changes every complete assignment by the same amount. The best assignment stays the same.

Reduced costs stay nonnegative. A complete matching of zeros reaches the lowest possible reduced cost, proving optimality.

If the zeros cannot form a full matching

Cover them with the fewest row/column lines. Let δ be the smallest uncovered value. Subtract δ from uncovered cells, add δ at line intersections, and repeat.

Hungarian algorithm: example

Illustrative costs. Rows = detections 1-3. Columns = template positions A-C.

1 Original costs

	A	B	C
1	4	1	3
2	2	0	5
3	3	2	2

Lower cost means a better match.

2 Row reduction

	A	B	C
1	3	0	2
2	2	0	5
3	1	0	0

Subtract row minima:
1, 0, 2.

3 Column reduction

	A	B	C
1	2	0	2
2	1	0	5
3	0	0	0

Subtract column minima:
1, 0, 0.

4 Adjustment

	A	B	C
1	1	0	1
2	0	0	4
3	0	1	0

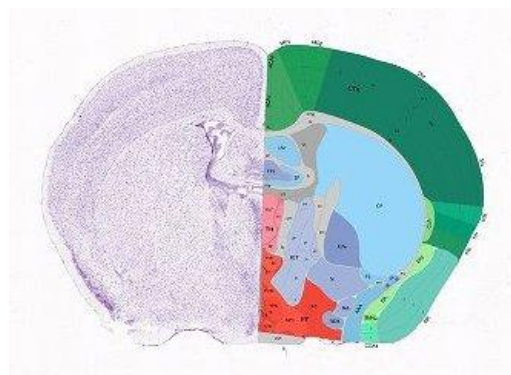
Cover row 3 + column B.
Smallest uncovered $\delta = 1$.

Why step 4 is needed: after column reduction, detections 1 and 2 both have a zero only in column B.

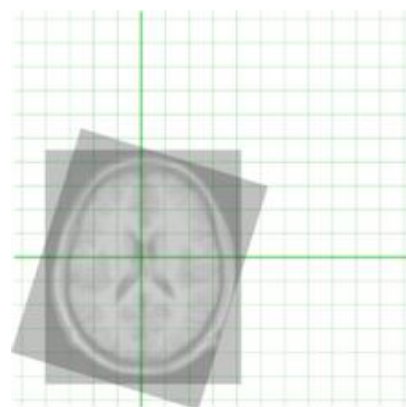
Optimal matching: 1 with B, 2 with A, 3 with C. Original total = 1 + 2 + 2 = 5.

Image registration

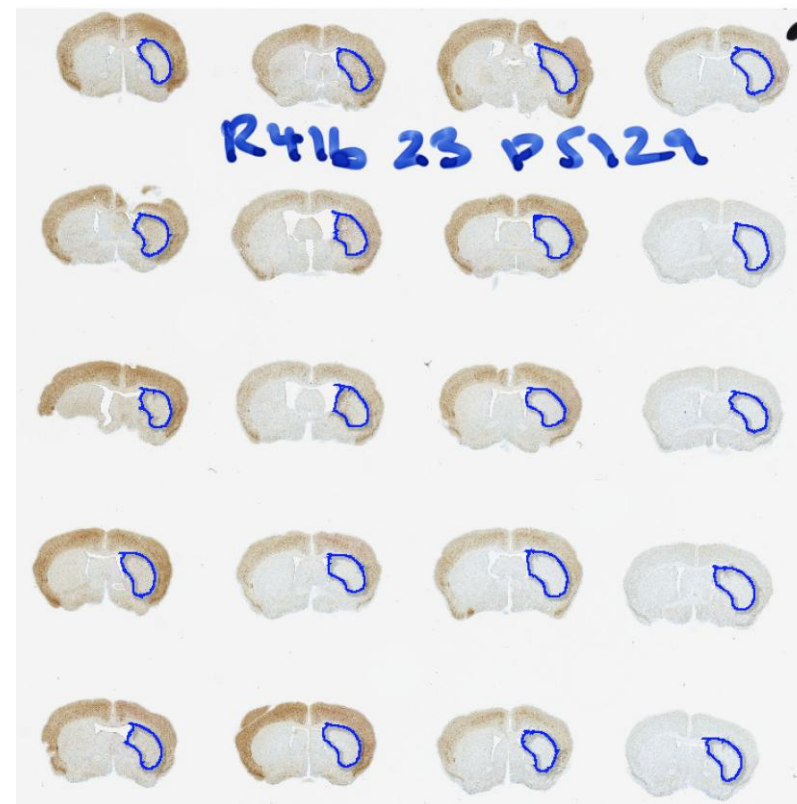
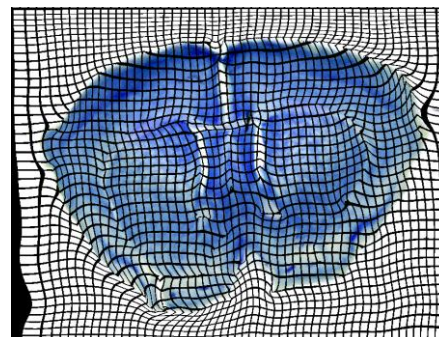
- Affine: NCC with Adam + AMSGrad
- Elastic: diffeomorphic cubic B-splines
- GPU fitting, full-resolution output



1 Affine: global alignment



2 Elastic: local refinement



Python libraries

AIRLAB
Autograd Image Registration Laboratory


PyTorch


NVIDIA

 **MSD**

Affine transformation

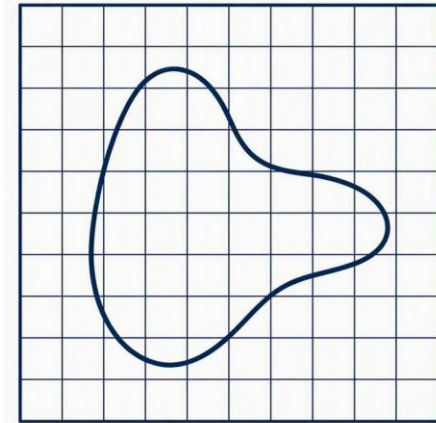
One global rule for every point

$$T(x) = Ax + t$$

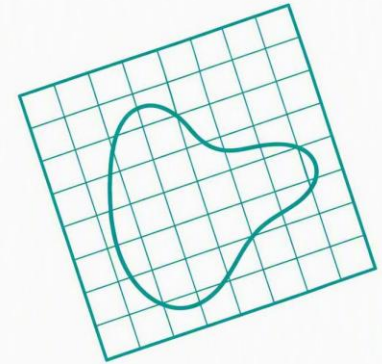
A controls rotation, scale and shear.
t shifts the image in two directions.

In 2D, a full affine transform has six parameters: four in A and two in t.

Global change in shape and position



Original grid



Affine transform

Straight lines remain straight. Parallel lines remain parallel.

Elastic transformation with B-splines

Local shifts vary across the image

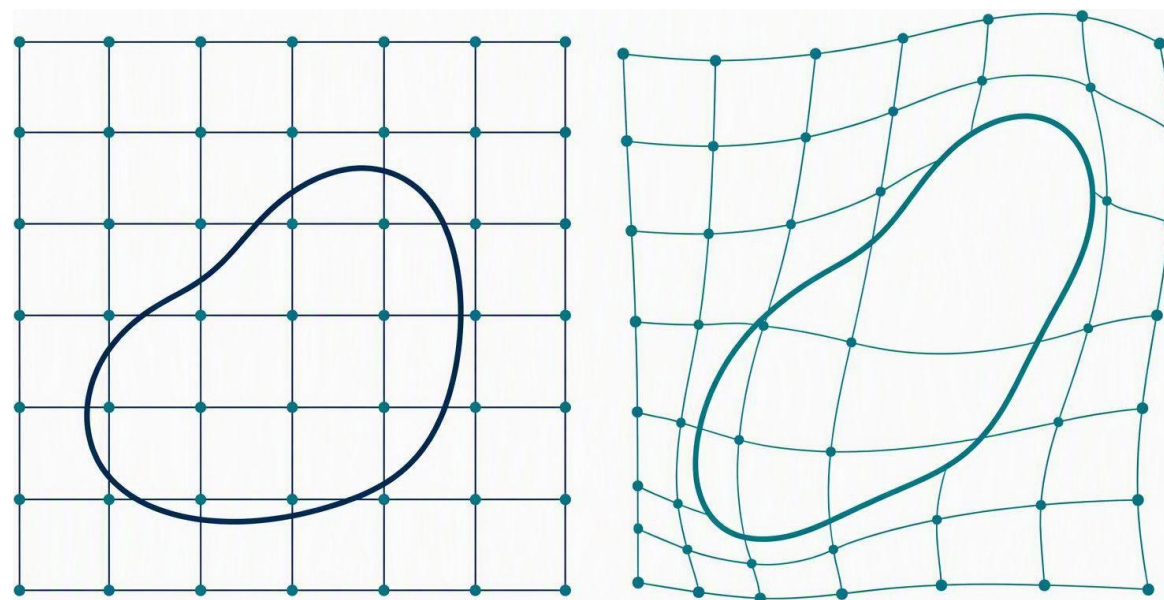
$$T(x) = x + u(x)$$

$u(x)$ describes the local displacement.

Nearby B-spline control points blend their influence to produce smooth motion.

After affine alignment, local deformation corrects remaining shape differences.

Smooth local bending



Regular grid

Local deformation

A smoothness penalty discourages abrupt changes between neighboring regions.

Approaches to finding the parameters

The transform defines what can move. The fitting method defines a good match.

Landmark-based

Match corresponding anatomical points. Fit the transform by minimizing distances between each pair.

Image-based

Compare pixel patterns. Optimize a similarity measure such as correlation or mutual information. NCC used

Hybrid

Combine image similarity with landmark errors. Weights control how strongly each guides the fit.

Affine parameters

A and t: six values in 2D

Elastic parameters

B-spline coefficients

Optimization of the transformation

A score guides parameter updates

$$E(\theta) = D(\theta) + \lambda R(\theta)$$

D measures mismatch between images or corresponding landmarks.

R penalizes rough local deformation.

λ controls the strength of that penalty.

θ contains A and t for affine fitting, or B-spline coefficients for elastic fitting.

How iterative fitting works

1 Initialize

Roughly align the images.

2 Compare

Warp the moving image and calculate E.

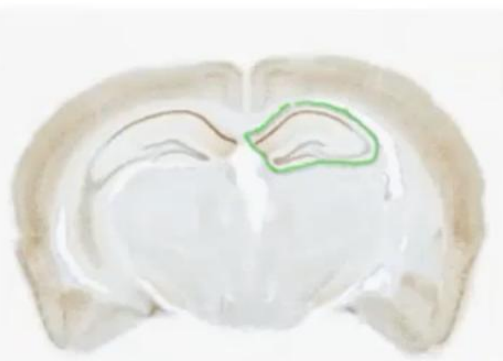
3 Update

Adjust parameters to reduce E, using an optimizer such as gradient descent or Adam.

4 Refine and inspect

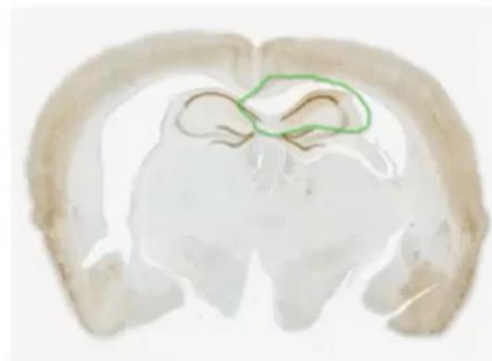
Repeat from coarse to fine resolution, then inspect anatomical alignment.

Results



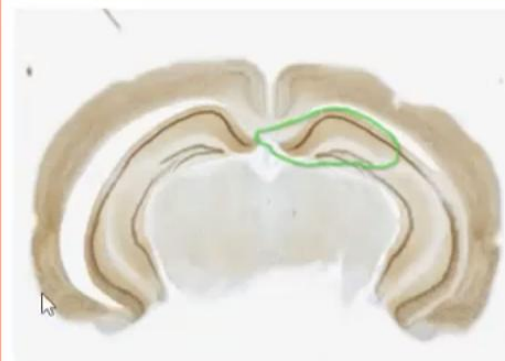
Sufficient boundaries

- Does not require manual intervention
- **66 %**



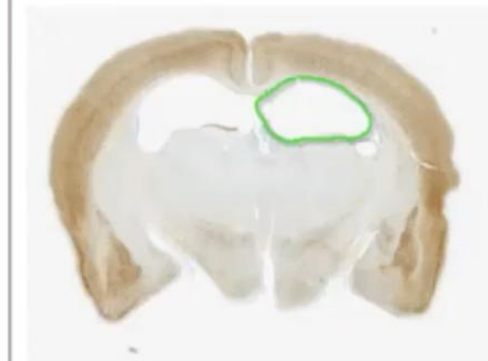
Inaccurate boundaries

- Technical processing artifacts
- Shifted brain midline
- Requires manual intervention
- **15 %**



Complete misalignment

- Inaccurate input
- Inaccurate identification of landmark boundaries
- Requires significant manual intervention
- **9 %**



Miscellaneous

- Acquisition artifacts
- Technical processing artifacts
- May be excluded from further evaluation
- **10 %**

Deployment and usage

- Deployment on internal HPC infrastructure
- Accessible via API
- Client-side GUI (local computers)
- Usage - semi-automated
- Quality control in place
- Missing alignments are corrected manually

Student opportunities at MSD

Part-time work alongside your studies

Practical experience

Contribute to real assignments and build professional experience.

Explore student roles in IT and other business functions.

Finding the right role

Look for student, intern and part-time vacancies.

Check the location, required skills and expected weekly hours.

Where to start

MSD Careers: jobs.msd.com/student-opportunities

Emerging Talent Rotational Program (ETR)

A full-time, entry-level start for recent graduates

3

IT-related rotations

2 years

program duration

Full-time

junior-level employment

Explore different IT roles

Rotations align development interests with business needs.

Build skills and a professional network.

Prague program details and applications: msd.cz/en/etr

Who should explore ETR?

Recent graduates with an interest in IT and healthcare.

Check the published eligibility criteria.

Thank you

