1

2 The ANTsX Ecosystem for Mapping the

3 Mouse Brain

4 Nicholas J. Tustison1, Min Chen2, Fae N. Kronman3, Jeffrey T. Duda2, Clare Gamlin4, Mia

5 G. Tustison, Michael Kunst4, Rachel Dalley4, Staci Sorenson4, Quanxi Wang4, Lydia Ng4,

6 Yongsoo Kim3, and James C. Gee2

7 1Department of Radiology and Medical Imaging, University of Virginia, Charlottesville, VA

8 2Department of Radiology, University of Pennsylvania, Philadelphia, PA

9 3Department of Neural and Behavioral Sciences, Penn State University, Hershey, PA

10 4Allen Institute for Brain Science, Seattle, WA

11

12 Corresponding author:

13 Nicholas J. Tustison, DSc

14 Department of Radiology and Medical Imaging

15 University of Virginia

16 [ntustison@virginia.edu](mailto:ntustison@virginia.edu)

17 **Abstract**

18 Precision mapping techniques coupled with high resolution image acquisition of the mouse

19 brain permit the study of the spatial organization of gene activity and their mutual interac-

20 tion for a comprehensive view of salient structural/functional relationships. Such research

21 is facilitated by standardized anatomical coordinate systems, such as the well-known Allen

22 Common Coordinate Framework (AllenCCFv3), and the ability to spatially map to such

23 standardized spaces. The Advanced Normalization Tools Ecosystem is a comprehensive

24 open-source software toolkit for generalized quantitative imaging with applicability to mul-

25 tiple organ systems, modalities, and animal species. Herein, we illustrate the utility of

26 ANTsX for generating precision spatial mappings of the mouse brain and potential sub-

27 sequent quantitation. We describe ANTsX-based workflows for mapping domain-specific

28 image data to AllenCCFv3 accounting for common artefacts and other confounds. Novel

29 contributions include ANTsX functionality for velocity flow-based mapping spanning the

30 spatiotemporal domain of a longitudinal trajectory which we apply to the Developmental

31 Common Coordinate Framework. Additionally, we present an automated structural morpho-

32 logical pipeline for determining volumetric and cortical thickness measurements analogous to

33 the well-utilized ANTsX pipeline for human neuroanatomical structural morphology which

34 illustrates a general open-source framework for tailored brain parcellations.

35 **1 Introduction**

36 Over the past two decades there have been significant advancements in mesoscopic analysis

37 of the mouse brain. It is currently possible to track single cell neurons in mouse brains,1

38 observe whole brain developmental changes on a cellular level,2 associate brain regions and

39 tissues with their genetic composition,3 and locally characterize neural connectivity.4 Much

40 of this scientific achievement has been made possible due to breakthroughs in high resolution

41 imaging techniques that permit submicron, 3-D imaging of whole mouse brains. Associated

42 research techniques such as micro-optical sectioning tomography,6 tissue clearing,1,7 spatial

43 transcriptomics9 are all well-utilized in the course of scientific investigations of mesoscale

44 relationships in the mouse brain.

45 An important component of this research is the ability to map the various image data to

46 anatomical reference frames11 for inferring spatial relationships between structures, cells,

47 and genetics. This has motivated the development of detailed structural image atlases of the

48 mouse brain. Notable examples include the Allen Brain Atlas and Common Coordinate Frameworks

49 (AllenCCFv3),13 the Waxholm Space,14 and more recently, the Developmental Common Co-

50 ordinate Framework (DevCCF).15 Despite the significance of these contributions, challenges

51 still exist in large part due to the wide heterogeneity in associated study-specific image

52 data. For example, variance in the acquisition methods can introduce artifacts such as tis-

53 sue distortion, holes, bubbles, folding, tears, and missing slices. These complicate assumed

54 correspondence for conventional spatial mapping approaches.

# 55 1.1 Mouse-specific brain mapping software

56 To address such challenges, several software packages have been developed over the years

57 comprising solutions of varying comprehensibility, sophistication, and availability. An

58 early contribution to the community was the Rapid Automatic Tissue Segmentation

59 (RATS) package16 for brain extraction. More recently, several publicly available packages

60 comprise well-established package dependencies originally developed on human brain data.

61 SPMMouse,17 for example, is based on the well-known Statistical Parametric Mapping

62 (SPM) Matlab-based toolset.18 The automated mouse atlas propagation (aMAP) tool is

63 largely a front-end for the NiftyReg image registration package19 applied to mouse data

64 which is currently available as a Python module.20 NiftyReg is also used by the Atlas-based

65 Imaging Data Analysis (AIDA) MRI pipeline21 as well as the Multi Atlas Segmentation

66 and Morphometric Analysis Toolkit (MASMAT). Whereas the former also incorporates the

67 FMRIB Software Library (FSL)22 for brain extraction and DSIStudio23 for DTI processing,

68 the latter uses NiftySeg and multi-consensus labeling tools24 for brain extraction and

69 parcellation. In addition, MASMAT incorporates N4 bias field correction25 from the

70 Advanced Normalization Tools Ecosystem (ANTsX)26 as do the packages Multi-modal

71 Image Registration And Connectivity anaLysis (MIRACL),27 Sammba-MRI,28 and Small

72 Animal Magnetic Resonance Imaging (SAMRI).29 However, whereas Saamba-MRI uses

73 AFNI30 for image registration; MIRACL, SAMRI, SAMBA,31 and BrainsMapi32 all use

74 ANTsX registration tools. Other packages use landmark-based approaches to image regis-

75 tration including SMART—33an R package for semi-automated landmark-based registration

76 and segmentation of mouse brain based on WholeBrain.34 FriendlyClearMap35 uses the

77 landmark-based registration functionality of Elastix.36 Finally, the widespread adoption

78 of deep learning techniques has also influenced development in mouse brain imaging

79 methodologies. For example, if tissue deformations are not considered problematic for a

80 particular dataset, DeepSlice can be used to determine affine mappings37 with the optimal

81 computational efficiency associated with neural networks.

# 82 1.2 The ANTsX Ecosystem for mouse brain mapping

83 As noted previously, many of the existing packages designed for processing mouse brain image

84 data use ANTsX tools for core processing steps in various workflows, particularly its pair-

85 wise, intensity-based image registration capabilities and bias field correction. Historically,

86 ANTsX development is originally based on fundamental approaches to image mapping,38–40

87 particularly in the human brain, which has resulted in core contributions to the field such as

88 the well-known Symmetric Normalization (SyN) algorithm.41 Since its development, various

89 independent platforms have been used to evaluate ANTsX image registration capabilities

Table 1: Sampling of ANTsX functionality

*ANTsPy: Preprocessing*

bias field correction n4\_bias\_field\_correction(...)

image denoising denoise\_image(...)

*ANTsPy: Registration*

image registration registration(...)

image transformation apply\_transforms(...)

template generation build\_template(...)

landmark registration fit\_transform\_to\_paired\_points(...)

time-varying landmark reg. fit\_time\_varying\_transform\_to\_point\_sets(...)

integrate velocity field integrate\_velocity\_field(...)

invert displacement field invert\_displacement\_field(...)

*ANTsPy: Segmentation*

MRF-based segmentation atropos(...)

Joint label fusion joint\_label\_fusion(...)

diffeormorphic thickness kelly\_kapowski(...)

*ANTsPy: Miscellaneous*

Regional intensity statistics label\_stats(...)

Regional shape measures label\_geometry\_measures(...)

B-spline approximation fit\_bspline\_object\_to\_scattered\_data(...)

Visualize images and overlays plot(...)

*ANTsPyNet: Mouse-specific*

brain extraction mouse\_brain\_extraction(...modality="t2"...) mouse\_brain\_extraction(...modality="ex5"...)

brain parcellation mouse\_brain\_parcellation(...)

cortical thickness mouse\_cortical\_thickness(...)

super resolution mouse\_histology\_super\_resolution(...)

ANTsX provides state-of-the-art functionality for processing biomedical image data. Such tools, including deep learning networks, support a variety of mapping-related tasks. A more comprehensive listing of ANTsX tools with self-contained R and Python examples is provided as a gist page on GitHub (<https://tinyurl.com/antsxtutorial>).

90 in the context of different application foci which include multi-site brain MRI data,42 pul-

91 monary CT data,43 and most recently, multi-modal brain registration in the presence of

92 tumors.44

93 Apart from its registration capabilities, ANTsX comprises additional functionality such

94 as template generation,45 intensity-based segmentation,46 preprocessing,25,47 deep learning

95 networks,26 and other miscelleneous utilties (see Table [1](#_bookmark0)). The comprehensive use of the

96 toolkit has demonstrated superb performance in multiple application areas (e.g., consensus

97 labeling,48 brain tumor segmentation,49 and cardiac motion estimation50 ). Importantly,

98 ANTs is built on the Insight Toolkit (ITK)51 deriving benefit from the open-source com-

99 munity of scientists and programmers and providing an important resource for algorithmic

100 development, evaluation, and improvement. We use this functionality to demonstrate re-

101 cently developed frameworks for mapping fluorescence micro-optical sectioning tomography

102 (fMOST) and multiplexed error-robust fluorescence in situ hybridization (MERFISH) im-

103 age data to the AllenCCFv3 atlas space. In addition to standard preprocessing steps (e.g.,

104 bias correction), additional considerations are accommodated within the ANTsX ecosystem,

105 such as section reconstruction and landmark-based alignment with corresponding processing

106 scripts available at <https://github.com/dontminchenit/CCFAlignmentToolkit>.

# 107 1.3 ANTsX-based open-source contributions

108 Consistent with previous ANTsX development, the newly introduced capabilities introduced

109 below are available through ANTsX (specifically, via R and Python ANTsX packages), and

110 illustrated through self-contained examples in the ANTsX tutorial ([https://tinyurl.com/](https://tinyurl.com/antsxtutorial)

111 [antsxtutorial](https://tinyurl.com/antsxtutorial)) with a dedicated GitHub repository specific to this work ([https://github.](https://github.com/ntustison/ANTsXMouseBrainMapping)

112 [com/ntustison/ANTsXMouseBrainMapping](https://github.com/ntustison/ANTsXMouseBrainMapping)).

## 113 1.3.1 The DevCCF velocity flow model

114 Recently, the Developmental Common Coordinate Framework (DevCCF) was introduced to

115 the mouse brain research community as a public resource15 comprising symmetric atlases of

116 multimodal image data and anatomical segmentations defined by developmental ontology.

117 These templates sample the mouse embryonic days (E) 11.5, E13.5, E15.5, E18.5 and postna-

118 tal day (P) 4, P14, and P56. Modalities include light sheet flourescence miscroscopy (LSFM)

119 and at least four MRI contrasts per developmental stage. Anatomical parcellations are also

120 available for each time point and were generated from ANTsX-based mappings of gene ex-

121 pression and other cell type data. Additionally, the P56 template was integrated with the

122 AllenCCFv3 to further increase the practical utility of the DevCCF. These processes, specif-

123 ically template generation and multi-modal image mapping, were performed using ANTsX

124 functionality in the presence of image mapping difficulties such as missing data and tissue

125 distortion.

126 Given the temporal gaps in the discrete set of developmental atlases, we also provide an

127 open-source framework for inferring correspondence within the temporally continuous do-

128 main sampled by the existing set of embryonic and postnatal atlases of the DevCCF. This

129 recently developed functionality permits the generation of a diffeomorphic velocity flow trans-

130 formation model,52 influenced by previous work.53 The resulting time-parameterized velocity

131 field spans the stages of the DevCCF where mappings between any two continuous time

132 points within the span bounded by the E11.5 and P56 atlases is determined by integration

133 of the optimized velocity field.

## 134 1.3.2 Structural morphology and cortical thickness in the mouse brain

135 One of the most frequently utilized pipelines in the ANTsX toolkit is that of estimating corti-

136 cal thickness maps in the human brain. Beginning with the Diffeomorphic Registration-based

137 Cortical Thickness (DiReCT) algorithm,54 this was later expanded to include a complete pro-

138 cessing framework for human brain cortical thickness estimation for both cross-sectional55

139 and longitudinal56 data using T1-weighted MRI. These pipelines were later significantly

140 refactored using deep learning innovations.26

141 In contrast to the pipeline development in human data,26 no current ANTsX tools exist to

142 create adequate training data for the mouse brain. In addition, mouse brain data acquisition

143 often has unique issues, such as lower data quality or sampling anisotropy which limits

144 its applicability to high resolution resources (e.g., AllenCCFv3, DevCCF), specifically with

145 respect to the corresponding granular brain parcellations derived from numerous hours of

146 expert annotation leveraging multimodal imaging resources.

147 Herein, we introduce a mouse brain cortical thickness pipeline for T2-weighted (T2-w) MRI com-

148 prising two novel deep learning components: two-shot learning brain extraction from data

149 augmentation of two ANTsX templates generated from two open datasets57,58 and single-

150 shot brain parcellation derived from the AllenCCFv3 labelings mapped to the corresponding

151 DevCCF P56 T2-w component. Although we anticipate that this cortical thickness pipeline

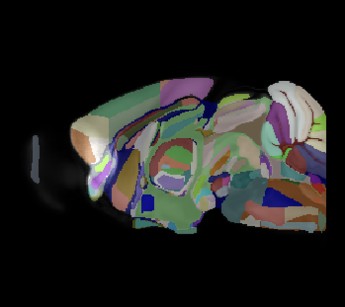
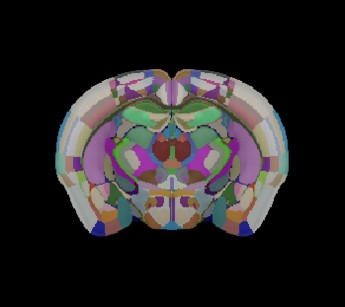
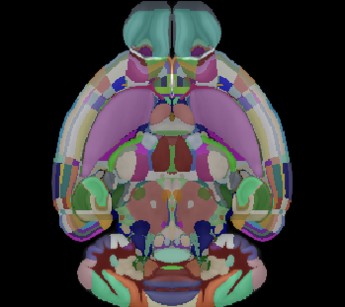
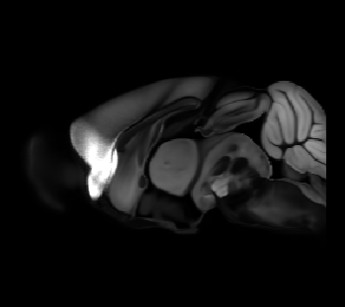
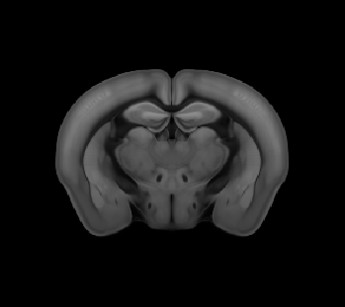
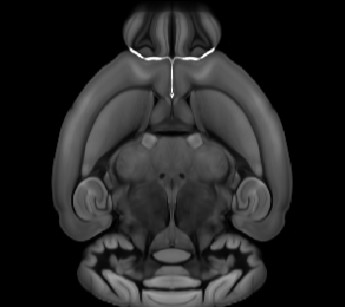
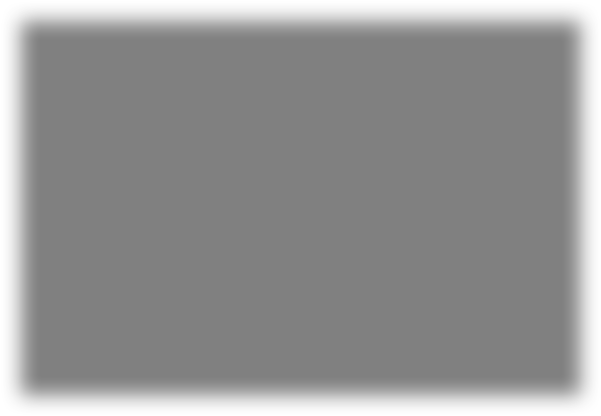
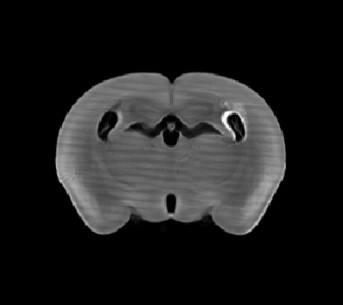
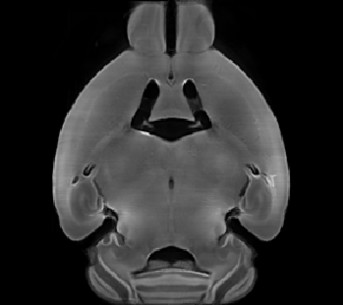
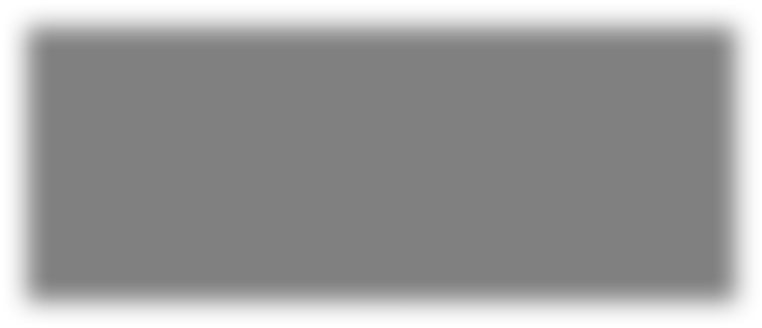
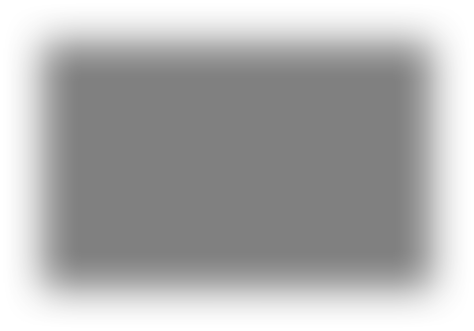
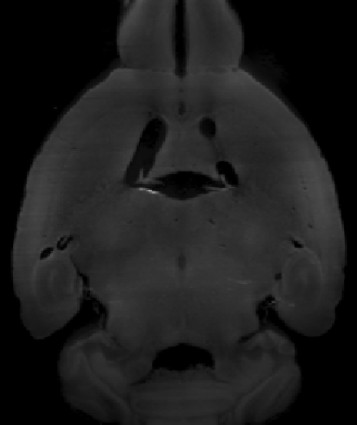
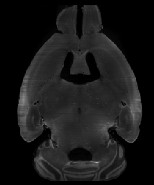
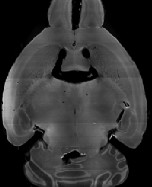
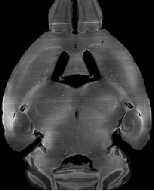
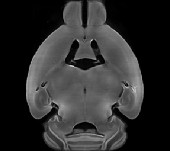
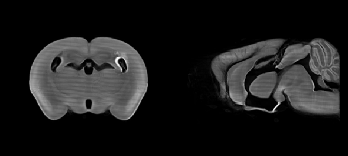
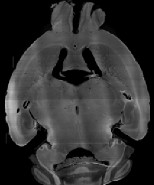
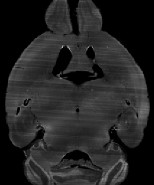
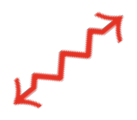
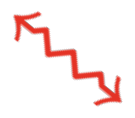
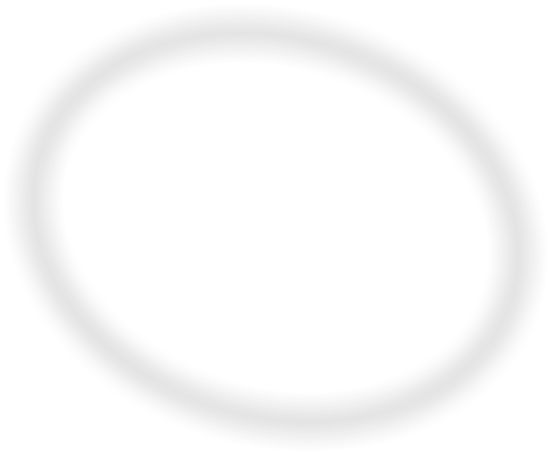
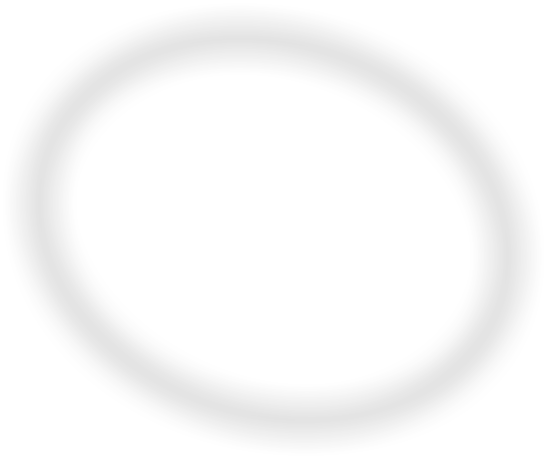
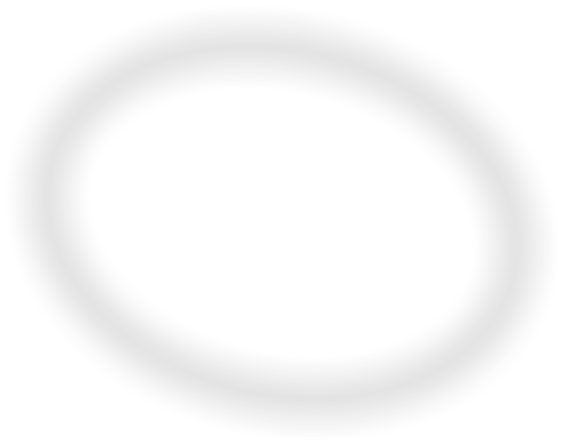
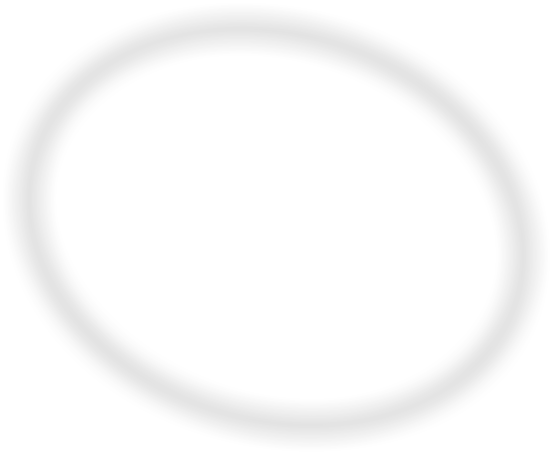
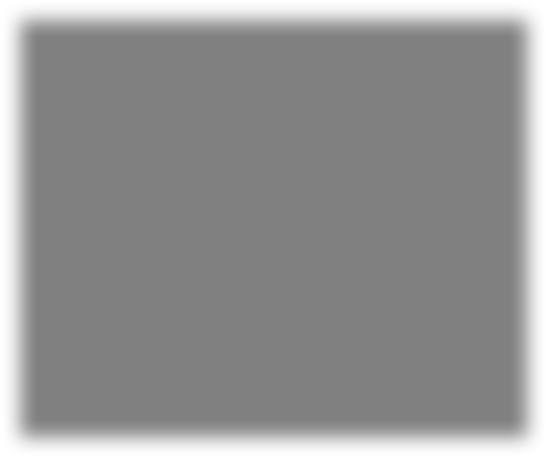
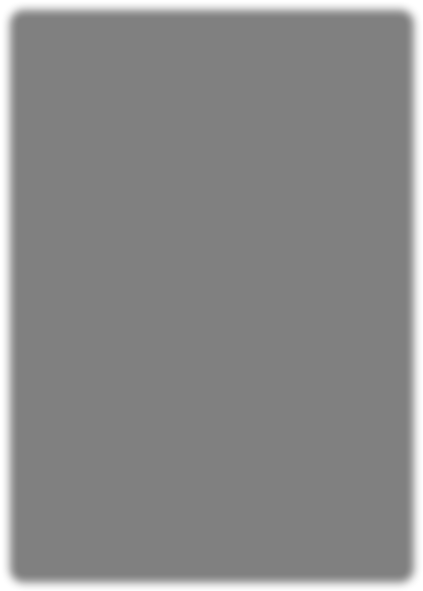
152 will be beneficial to the research community, this work demonstrates more generally how

153 one can leverage ANTsX tools for developing tailored brain parcellation schemes using these

154 publicly available resources. Evaluation is performed on an independent open data set59

155 comprising longitudinal acquisitions of multiple specimens.

156 **2 Results**



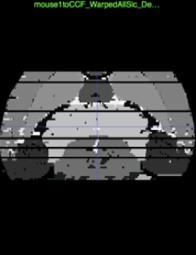
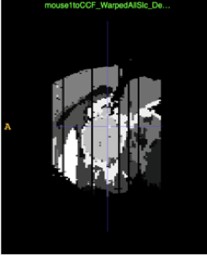
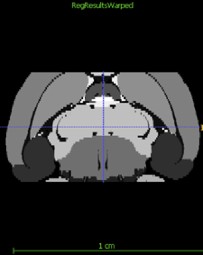
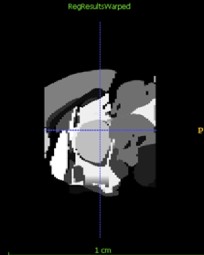
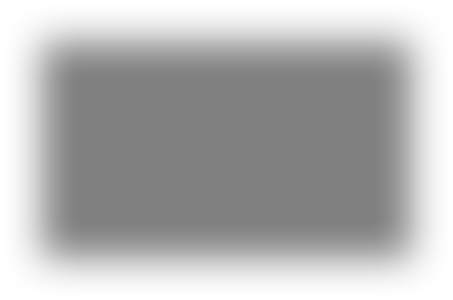
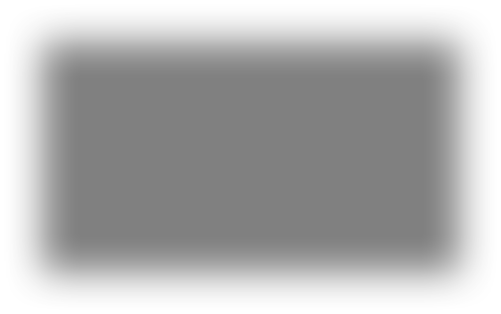
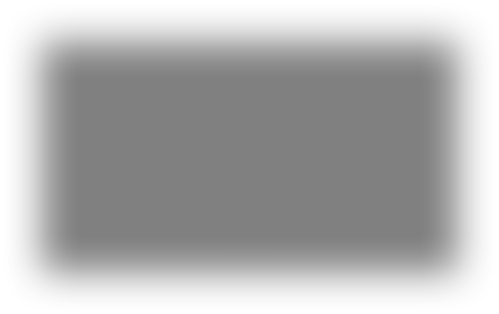
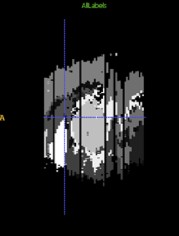
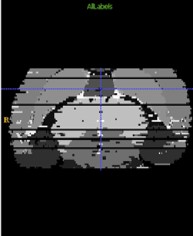
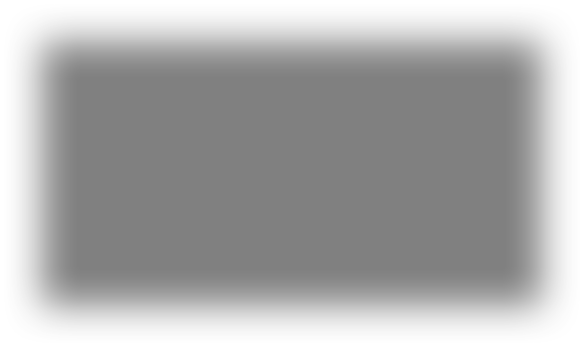
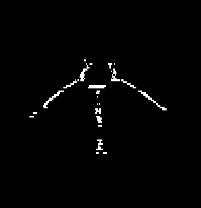
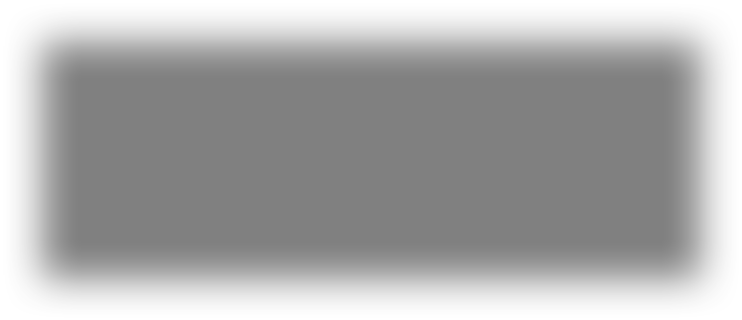
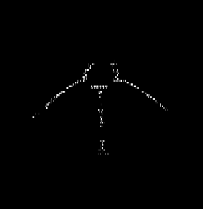
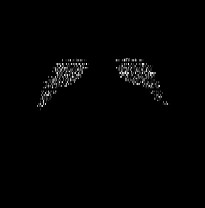
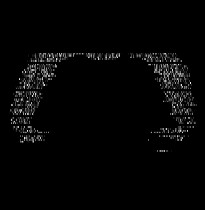
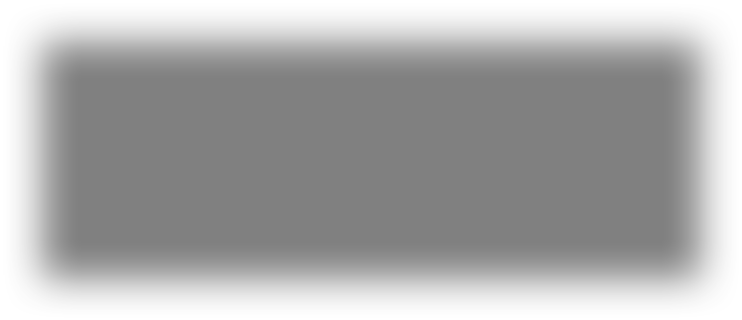
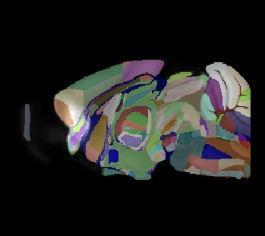
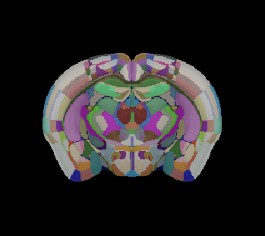
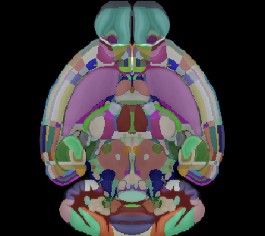
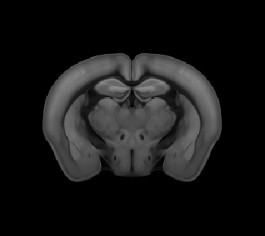
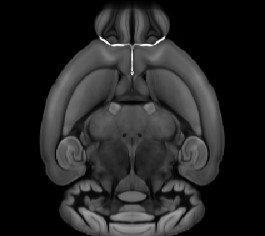
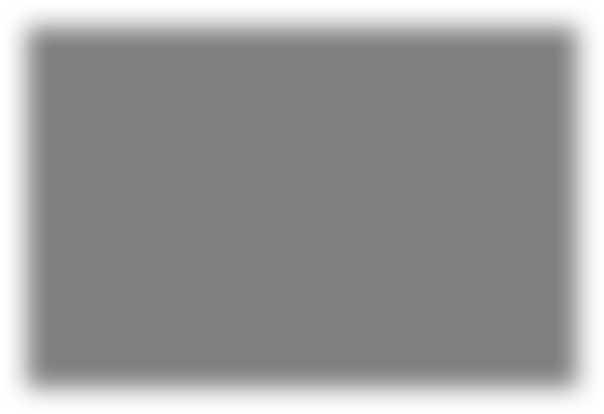
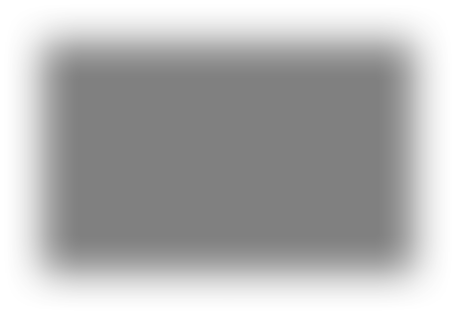
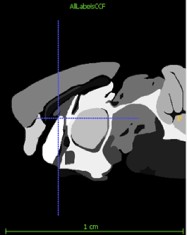
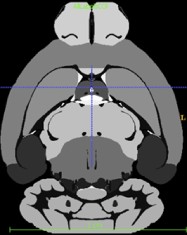
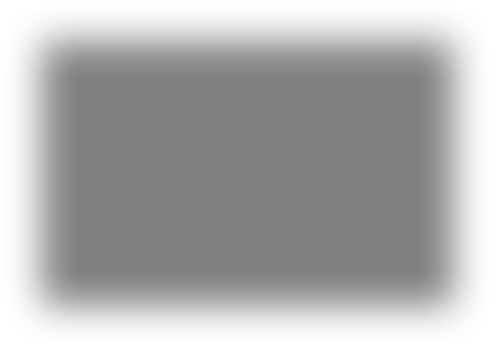
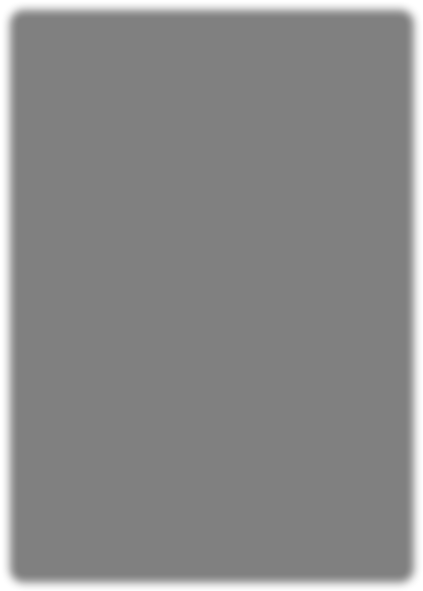
Input fMOST

N4 bias correction

**AllenCCFv3**

**fMOST template**

**fMOST template generation**



Input MERFISH

**AllenCCFv3**

**MERFISH/AllenCCFv3 mapping**

Conjoin anatomical labels

Section and FOV

matched

Ontological MERFISH-based simplification

MERFISH

anatomical labeling

(a) (b)

Figure 1: Diagrammatic illustration of the two ANTsX-based pipelines for mapping (a) fMOST and (b) MERFISH data into the space of AllenCCFv3. Each generates the requisite transforms, *T* , to map individual images.

# 157 2.1 AllenCCFv3 brain image mapping

## 158 2.1.1 Mapping fluorescence micro-optical sectioning tomography (fMOST) data

159 **Overview.** A framework for mapping fluorescence micro-optical sectioning tomography

160 (fMOST) mouse brain images into the AllenCCFv3 was developed (see Figure [1](#_bookmark1)(a)). An

161 intensity- and shape-based average fMOST atlas serves as an intermediate registration target

162 for mapping fMOST images from individual specimens into the AllenCCFv3. Preprocess-

163 ing steps include downsampling to match the 25*µm* isotropic AllenCCFv3, acquisition-based

164 stripe artifact removal, and inhomogeneity correction.25 Preprocessing also includes a single

165 annotation-driven registration to establish a canonical mapping between the fMOST atlas

166 and the AllenCCFv3. This step allows us to align expert determined landmarks to accu-

167 rately map structures with large morphological differences between the modalities, which are

168 difficult to address using standard approaches. Once this canonical mapping is established,

169 standard intensity-based registration is used to align each new fMOST image to the fMOST

170 specific atlas. This mapping is concatenated with the canonical fMOST atlas-to-AllenCCFv3

171 mapping to further map each individual brain into the latter without the need to generate

172 additional landmarks. Transformations learned through this mapping can be applied to sin-

173 gle neuron reconstructions from the fMOST images to evaluate neuronal distributions across

174 different specimens into the AllenCCFv3 for the purpose of cell census analyses.

175 **Data.** The high-throughput and high-resolution fluorescence micro-optical sectioning to-

176 mography (fMOST)60,61 platform was used to image 55 mouse brains containing gene-defined

177 neuron populations, with sparse transgenic expression.62,63 In short, the fMOST imaging

178 platform results in 3D images with voxel sizes of 0*.*35 *×* 0*.*35 *×* 1*.*0 *µm*3 and is a two-channel

179 imaging system where the green channel displays the GFP labeled neuron morphology and

180 the red channel is used to visualize the counterstained propidium iodide cytoarchitecture.

181 The spatial normalizations described in this work were performed using the red channel,

182 which offered higher tissue contrast for alignment, although other approaches are possible

183 including multi-channel registration.

184 **Evaluation.** Evaluation of the canonical fMOST atlas to Allen CCFv3 mapping was per-

185 formed via quantitative comparison at each step of the registration and qualitative assess-

186 ment of structural correspondence after alignment by an expert anatomist. Dice values were

187 generated for the following structures: whole brain, 0.99; fimbria, 0.91; habenular commissure,

188 0.63; posterior choroid plexus, 0.93; anterior choroid plexus, 0.96; optic chiasm, 0.77; cau-

189 date putamen, 0.97. Similar qualitative assessment was performed for each fMOST specimen

190 including the corresponding neuron reconstruction data.

## 191 2.1.2 Mapping multiplexed error-robust fluorescence in situ hybridization

192 **(MERFISH) data**

193 **Overview.** The unique aspects of mapping multiplexed error-robust fluorescence in situ

194 hybridization (MERFISH) spatial transcriptomic data onto AllenCCFv364 required the de-

195 velopment of a separate ANTsX-based pipeline (see Figure [1](#_bookmark1)(b)). Mappings are performed

196 by matching gene expression derived region labels from the MERFISH data to corresponding

197 anatomical parcellations of the AllenCCFv3. The pipeline consists of MERFISH data spe-

198 cific preprocessing which includes section reconstruction, mapping corresponding anatomical

199 labels between AllenCCFv3 and the spatial transcriptomic maps of the MERFISH data, and

200 matching MERFISH sections to the atlas space. Following pre-processing, two main align-

201 ment steps were performed: 1) 3D global affine mapping and section matching of the Al-

202 lenCCFv3 into the MERFISH data and 2) 2D global and deformable mapping between each

203 MERFISH section and matched AllenCCFv3 section. Mappings learned via each step in the

204 pipeline are preserved and concatenated to provide point-to-point correspondence between

205 the original MERFISH data and AllenCCFv3, thus allowing individual gene expressions to

206 be transferred into the AllenCCFv3.

207 **Data.** MERFISH mouse brain data was acquired using the procedure detailed in 64 Briefly,

208 a brain of C57BL/6 mouse was dissected according to standard procedures and placed into

209 an optimal cutting temperature (OCT) compound (Sakura FineTek 4583) in which it was

210 stored at -80 °C. The fresh frozen brain was sectioned at 10 *µm* on Leica 3050 S cryostats at

211 intervals of 200 *µm* to evenly cover the brain. A set of 500 genes were imaged that had been

212 carefully chosen to distinguish the *∼*5200 clusters of our existing RNAseq taxonomy. For

213 staining the tissue with MERFISH probes, a modified version of instructions provided by

214 the manufacturer was used.64 Raw MERSCOPE data were decoded using Vizgen software

215 (v231). Cell segmentation was performed.65 In brief, cells were segmented based on DAPI and

216 PolyT staining using Cellpose.66 Segmentation was performed on a median z-plane (fourth

217 out of seven) and cell borders were propagated to z-planes above and below. To assign

218 cluster identity to each cell in the MERFISH dataset, we mapped the MERFISH cells to the

219 scRNA-seq reference taxonomy.

220 **Evaluation.** Alignment of the MERFISH data into the AllenCCFv3 was qualitatively as-

221 sessed by an expert anatomist at each iteration of the registration using known correspon-

222 dence of gene markers and their associations with the AllenCCFv3. As previously reported,64

223 further assessment of the alignment showed that of the 554 terminal regions (gray matter only) in

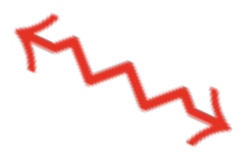
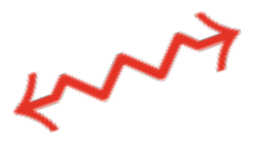
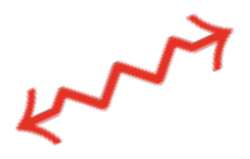
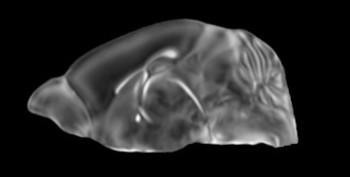
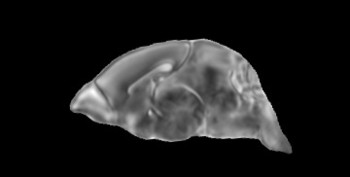
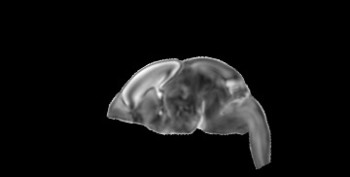
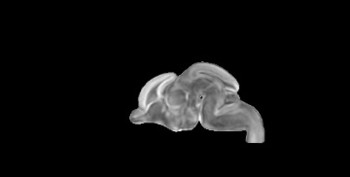
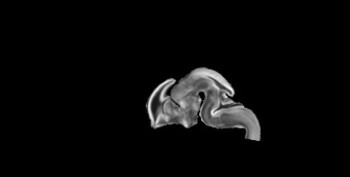
224 the AllenCCFv3, only seven small subregions were missed from the MERFISH dataset:

225 frontal pole, layer 1 (FRP1), FRP2/3, FRP5; accessory olfactory bulb, glomerular layer

226 (AOBgl); accessory olfactory bulb, granular layer (AOBgr); accessory olfactory bulb, mitral

227 layer (AOBmi); and accessory supraoptic group (ASO).

# 228 2.2 The DevCCF velocity flow model



**Developmental CCF velocity flow model**

**E18.5**

**E15.5**

**P04**

**P56**

**E11.5**

**E13.5**

**P14**

Figure 2: The spatial transformation between any two time points within the DevCCF longitudinal developmental trajectory is available through the use of ANTsX functionality for generating a velocity flow model.

229 To continuously interpolate transformations between the different stages of the DevCCF

230 atlases, a velocity flow model was constructed using DevCCF derived data and functionality

231 recently introduced into both the ANTsR and ANTsPy packages. Both platforms include

232 a complete suite of functions for determining dense correspondence from sparse landmarks

233 based on a variety of transformation models ranging from standard linear models (i.e., rigid,

234 affine) to deformable diffeomorphic models (e.g, symmetric normalization).41 The latter set

235 includes transformation models for both the pairwise scenario and for multiple sets, as in the

236 case of the DevCCF. ANTsX, being built on top of ITK, uses an ITK image data structure

237 for the 4-D velocity field where each voxel contains the *x*, *y*, *z* components of the field at

238 that point.

239 **2.2.1 Data**

240 Labeled annotations are available as part of the original DevCCF and reside in the space

241 of each developmental template which range in resolution from 31*.*5 *–* 50 *µ*m. Across all

242 atlases, the total number of labeled regions exceeds 2500. From these labels, a common set

243 of 26 labels (13 per hemisphere) across all atlases were used for optimization and evaluation.

244 These simplified regions include: terminal hypothalamus, subpallium, pallium, peduncular

245 hypothalamus, prosomere, prosomere, prosomere, midbrain, prepontine hindbrain, pontine

246 hindbrain, pontomedullary hindbrain, medullary hindbrain, and tracts.

247 Prior to velocity field optimization, all data were rigidly transformed to DevCCF P56 using

248 the centroids of the common label sets. In order to determine the landmark correspondence

249 across DevCCF stages, the multi-metric capabilities of ants.registration(...) were used.

250 Instead of performing intensity-based pairwise registration directly on these multi-label im-

251 ages, each label was used to construct a separate fixed and moving image pair resulting in a

252 multi-metric registration optimization scenario involving 24 binary image pairs (each label

253 weighted equally) for optimizing diffeomorphic correspondence between neighboring time

254 point atlases using the mean squares metric and the symmetric normalization transform.

255 To generate the set of common point sets across all seven developmental atlases, the label

256 boundaries and whole regions were sampled in the P56 atlas and then propagated to each

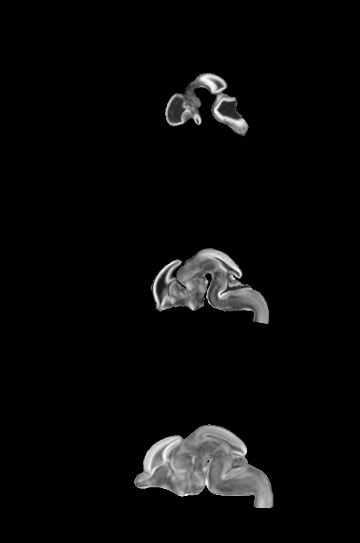
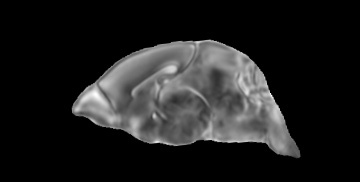
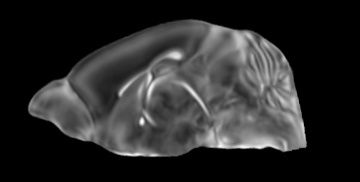
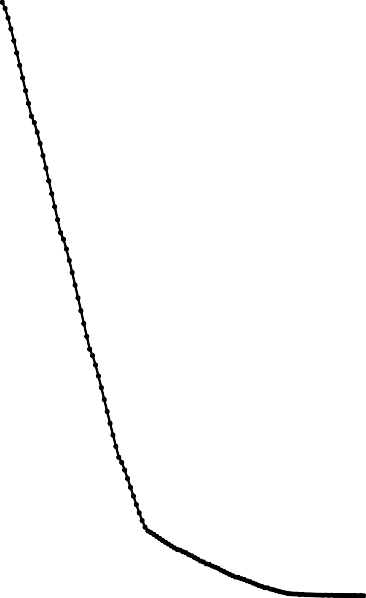
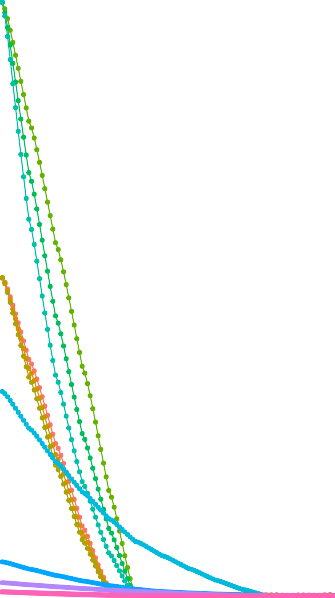
257 atlas using the transformations derived from the pairwise registrations. We selected a sam-

258 pling rate of 10% for the contour points and 1% for the regional points for a total number

259 of points per atlas being 173303 (*Ncontour* = 98151 and *Nregion* = 75152). Regional

260 boundary points were weighted twice as those of non-boundary points during optimization.

0.4



**Overall convergence**

**Per integration point**

**E11.5**

**1.0**

0.100

**0.9**

**E13.5**

**0.8**

**E15.5**

0.075

**0.7**

**E18.5**

**0.6**

**P04**

**0.5**

0.050

**0.4**

**0.3**

**P14**

0.025

**0.2**

**0.1**

**P56**

0.000

**0.0**

0

25

50

75 100 125

0

25 50 75 100 125

**Iteration Iteration**

***t* = 0.66**

***t* = 0.79 *t* = 0.89 *t* = 1.0**

0.2

Integration point 1

Integration point 2

Integration point 3

Integration point 4

Integration point 5

Integration point 6

Integration point 7

Integration point 8

Integration point 9

Integration point 10

Integration point 11

**Total distance error (mm)**

**Median distance error (mm)**

**Integration point**

***t* = 0.25**

***t* = 0.51**

*Logarithmically distributed*

0.0

***t* = 0.0**

Figure 3: Convergence of the optimization of the velocity field for describing the transfor- mation through the developmental stages from E11.5 through P56.

## 261 2.2.2 Optimization

262 The velocity field was optimized using the input composed of the seven corresponding point

263 sets and their associated weight values, the selected number of integration points for the

264 velocity field (*N* = 11), and the parameters defining the geometry of the spatial dimensions

265 of the velocity field. Thus, the optimized velocity field described here is of size [256*,* 182*,* 360]

266 (50 *µ*m isotropic) *×*11 integration points for a total compressed size of a little over 2 GB.

267 This choice represented weighing the trade-off between tractability, portability, and accuracy.

268 However, all data and code to reproduce the results described (with possible variation in the

269 input parameters) are available in the dedicated GitHub repository.

270 The normalized time point scalar value for each atlas/point-set in the temporal domains [0*,* 1]

271 was also defined. Given the increasingly larger gaps in the postnatal timepoint sampling, we

272 made two adjustments. Based on known mouse brain development, we used 28 days for the

273 P56 data. We then computed the log transform of the adjusted set of time points prior to

274 normalization between 0 and 1 (see the right side of Figure [3](#_bookmark2)). This log transform, as part

275 of the temporal normalization, significantly improved data spacing.

276 The maximum number of iterations was set to 200 with each iteration taking six minutes. At each

277 iteration we looped over the 11 integration points. At each integration point, the velocity field

278 estimate was updated by warping the two immediately adjacent point sets to the integration

279 time point and determining the regularized displacement field between the two warped point

280 sets. As with any gradient-based descent algorithm, this field was multiplied by a small step

281 size (*δ* = 0*.*2) before adding to the current velocity field. Convergence is determined by the

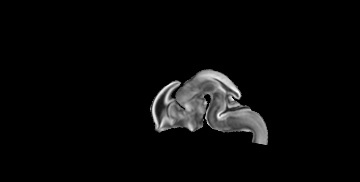
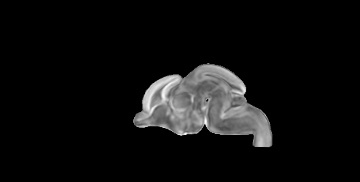
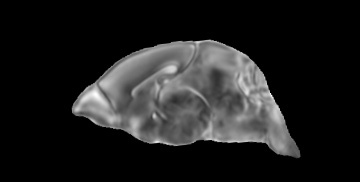
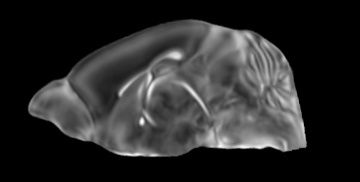
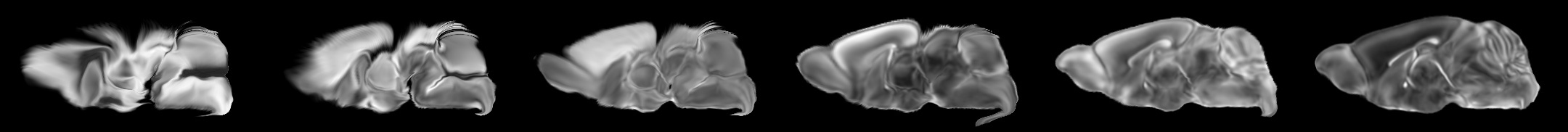
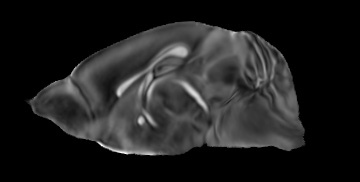
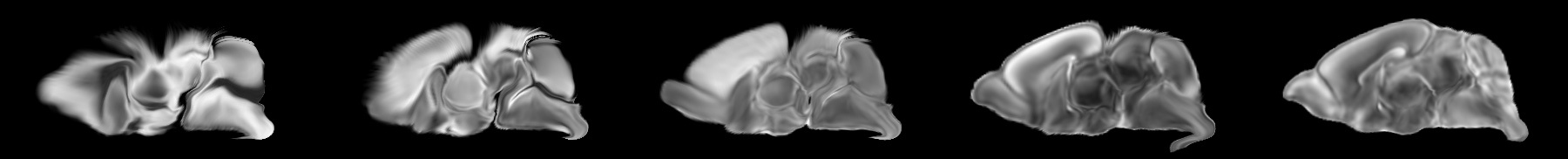
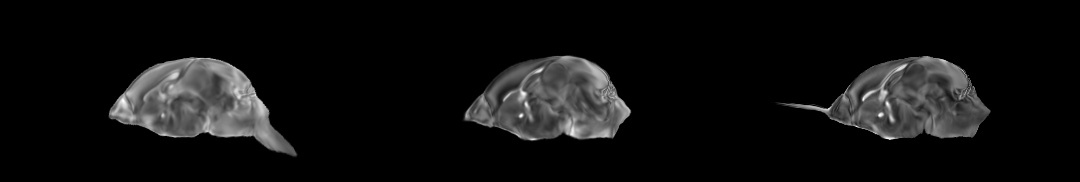
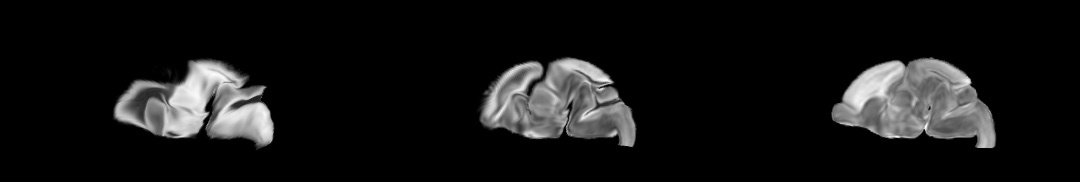
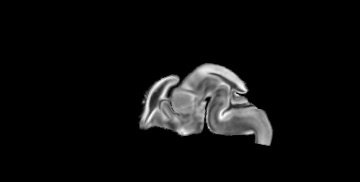
282 average displacement error over each of the integration points. As can be seen in the left

283 panel of Figure [3](#_bookmark2), convergence occurred around 125 iterations when the average displacement

284 error over all integration points is minimized. The median displacement error at each of the

285 integration points also trends towards zero but at different rates.

## 286 2.2.3 The transformation model



**P56**

**P14**

**P04**

**E18.5**

**E15.5**

**E13.5**

**E11.5**

Figure 4: Mid-sagittal visualization of the effects of the transformation model in warping every developmental stage to the time point of every other developmental stage. The original images are located along the diagonal. Columns correspond to the warped original image whereas the rows represent the reference space to which each image is warped.

287 Once optimized, the resulting velocity field can be used to generate the deformable transform

288 between any two continuous points within the time interval bounded by E11.5 and P56. In

289 Figure [4](#_bookmark3), we transform each atlas to the space of every other atlas using the DevCCF

290 transform model. Additionally, one can use this transformation model to construct virtual

291 templates in the temporal gaps of the DevCCF. Given an arbitrarily chosen time point

292 within the normalized time point interval, the existing adjacent DevCCF atlases on either

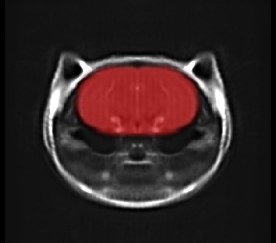
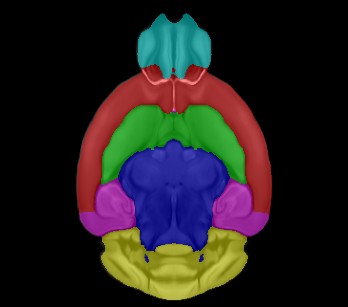
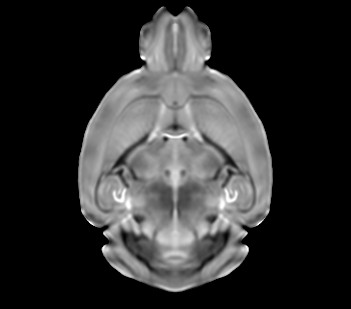
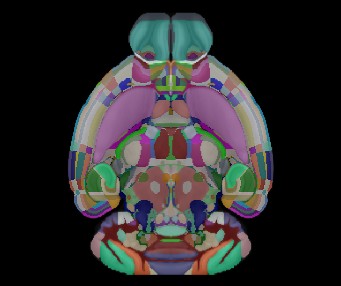
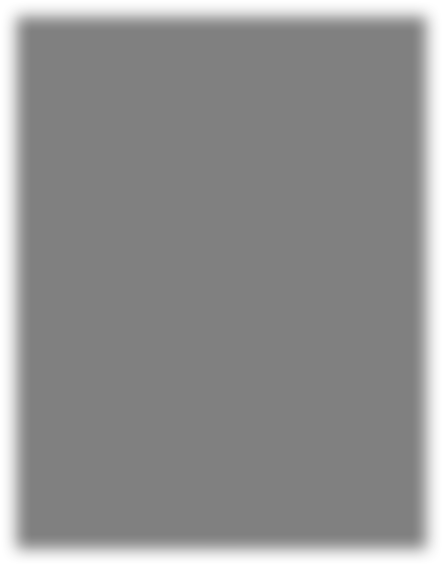
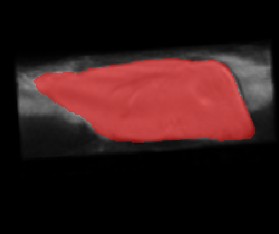
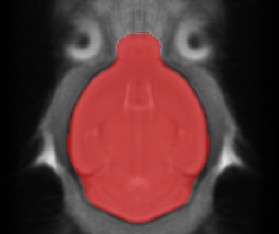
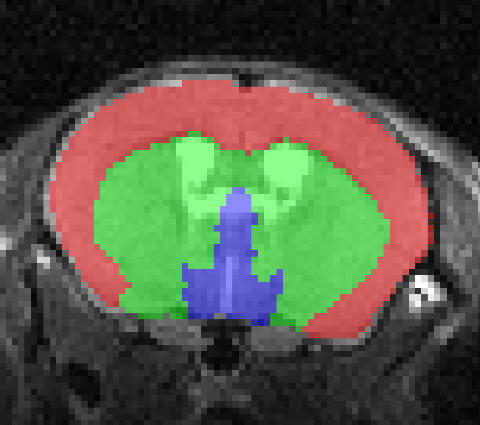
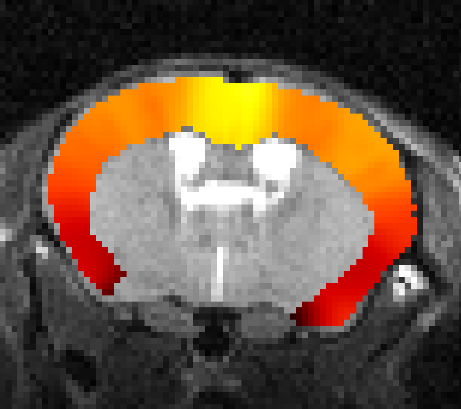
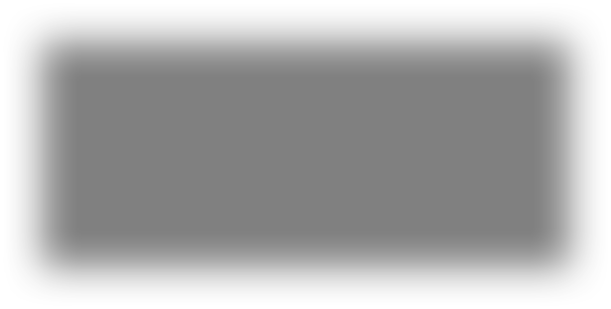
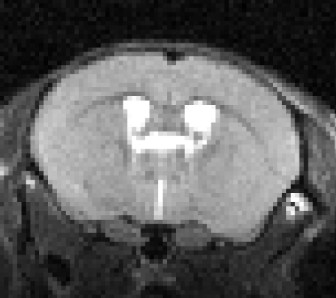
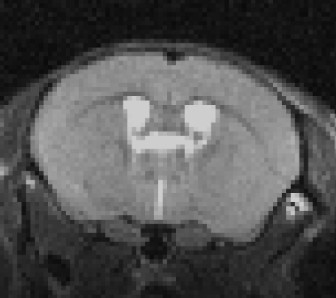
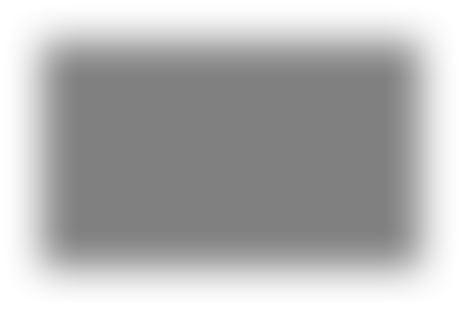
293 chronological side can be warped to the desired time point. A subsequent call to one of the

294 ANTsX template building functions then permits the construction of the template at that

295 time point. Note that both of these usage examples can be found in the GitHub repository

296 previously given.

# 297 2.3 The Mouse Cortical Thickness Pipeline



**Brain parcellation**

Input MRI

N4 bias correction

KellyKapowski cortical thickness

Ontological simplification and warping to DevCCF

AllenCCFv3

**Brain extraction**

Data augmentation/ network training

DevCCF T2-w with labels

Data augmentation/ network training

B-spline HR template

CAMRI template

Figure 5: The mouse brain cortical thickness pipeline integrating two deep learning compo- nents for brain extraction and brain parcellation prior to estimating cortical thickness. Both deep learning networks rely heavily on data augmentation on templates built from open data and provide an outline for further refinement and creating alternative parcellations for tailored research objectives.

298 One of the most well-utilized pipelines in the ANTsX toolkit is the generation of corti-

299 cal thickness maps in the human brain from T1-weighted MRI. Starting with the novel

300 Diffeomorphic Registration-based Cortical Thickness (DiReCT) algorithm,54 a complete al-

301 gorithmic workflow was developed for both cross-sectional55 and longitudinal56 T1-weighted

302 MR image data. This contribution was later refactored using deep learning26 leveraging the

303 earlier results55 for training data.

304 In the case of the mouse brain, the lack of training data and/or tools to generate training

305 data make a similar developmental trajectory difficult. In addition, mouse data is often

306 characterized by unique issues such as frequent anisotropic sampling which are often in sharp

307 contrast to the high resolution resources available within the community, e.g., AllenCCFv3

308 and DevCCF. Using ANTsX and other publicly available data resources, we developed a

309 complete mouse brain structural morphology pipeline as illustrated in Figure [5](#_bookmark4) and detailed

310 below.

## 311 2.3.1 Two-shot mouse brain extraction network

312 In order to create a generalized mouse brain extraction network, we built whole-head tem-

313 plates from two publicly available datasets. The Center for Animal MRI (CAMRI) dataset57

314 from the University of North Carolina at Chapel Hill consists of 16 T2-weighted MRI volumes of voxel resolution 0*.*16 *×* 0*.*16 *×* 0*.*16 *mm*3. The

315 second high-resolution data set58 comprises 88 specimens each with three spatially aligned

316 canonical views with in-plane resolution of 0*.*08 *×* 0*.*08 *mm*2 with a slice thickness of 0*.*5 *mm*.

317 These three orthogonal views were used to reconstruct a single high-resolution volume per

318 subject using a B-spline fitting algorithm developed in ANTsX.67 From these two datasets,

319 two symmetric isotropic ANTsX templates45 were generated having different defacing aes-

320 thetics analogous to the publicly available ANTsX human brain templates used in previ-

321 ous research.55 Bias field simulation, intensity histogram warping, noise simulation, random

322 translation and warping, and random anisotropic resampling in the three canonical directions

323 were used for data augmentation in creating a T2-weighted brain extraction network.

## 324 2.3.2 Single-shot mouse brain parcellation network

325 To create the network for generating a brain parcellation consistent with cortical thickness

326 estimation, we used the AllenCCFv3 and the associated allensdk Python library. Using

327 allensdk, a gross parcellation labeling was generated from the fine Allen CCFv3 labeling

328 which includes the cerebral cortex, cerebral nuclei, brain stem, cerebellum, main olfactory

329 bulb, and hippocampal formation. This labeling was mapped to the P56 component of the

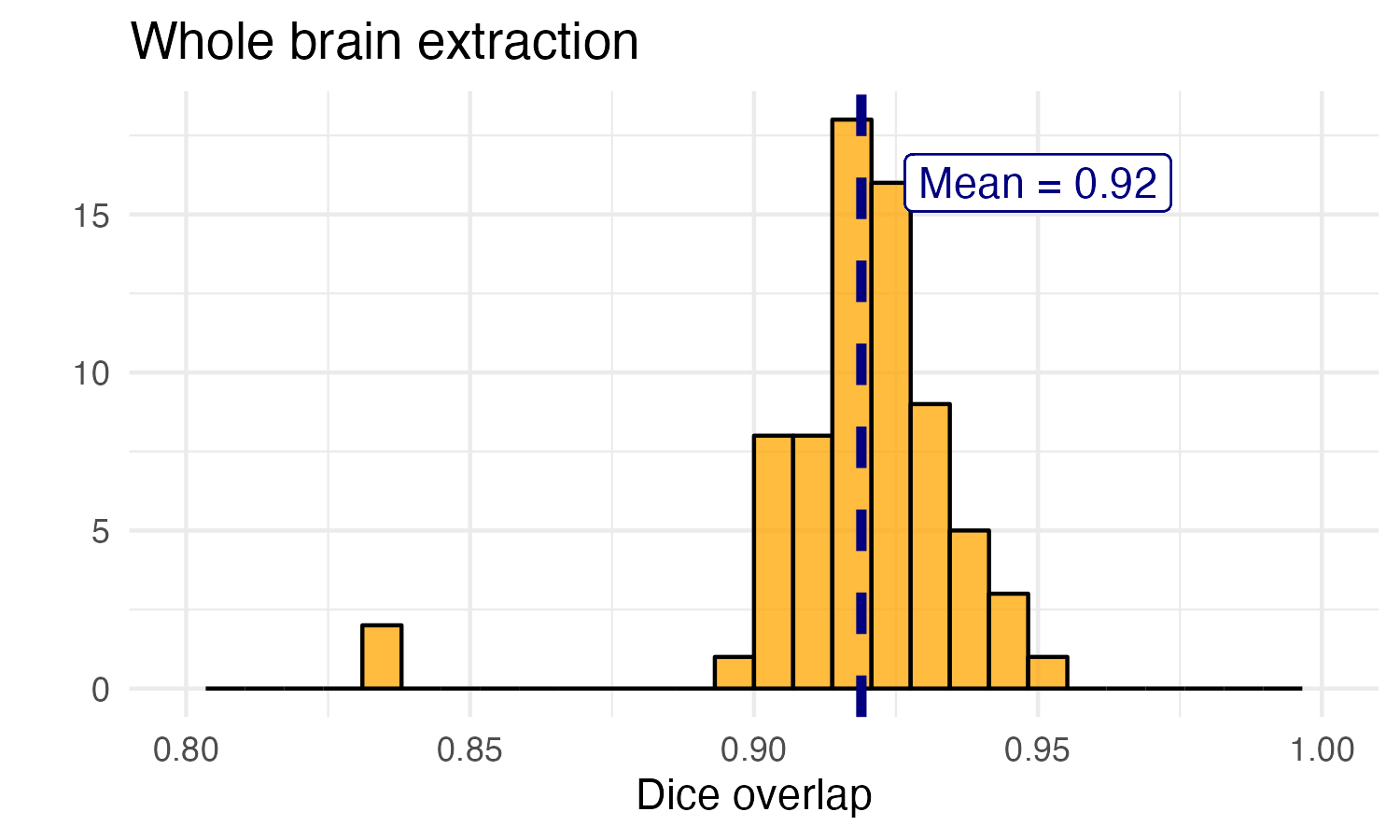
330 DevCCF. Both the T2-w P56 DevCCF and labelings, in conjunction with the data aug-

331 mentation described previously for brain extraction, was used to create a brain parcellation

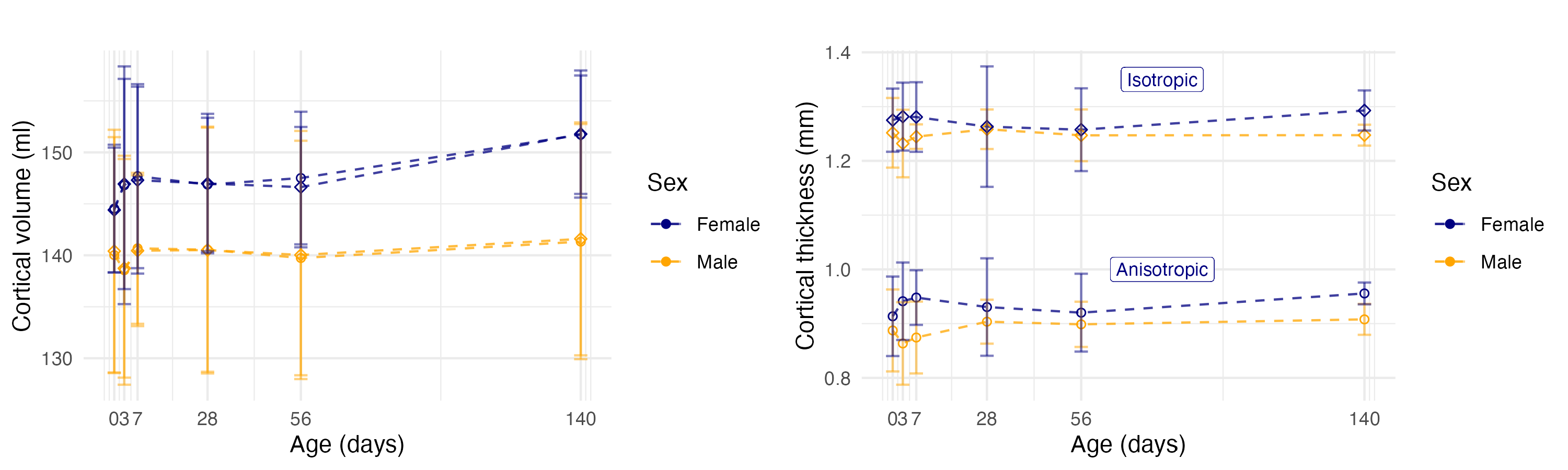
332 network.

333

## 2.3.3 Evaluation



(a)



(b) (c)

Figure 6: Evaluation of the ANTsX mouse brain extraction, parcellation, and cortical thick- ness pipeline on an independent dataset consisting of 12 specimens *×* 7 time points = 84 total images. (a) Dice overlap comparisons with the provided brain masks provide generally good agreement with the brain extraction network. (b) Cortical volume measurements show sim- ilar average quantities over the developmental trajectory between the original anisotropic data and interpolated isotropic data. (c) These results contrast with the cortical thickness measurements which show that cortical thickness estimation in anisotropic space severely underestimates the actual values.

334 For evaluation, we used an additional publicly available dataset59 which is completely in-

335 dependent from the data used in training the brain extraction and parcellation networks.

336 Data includes 12 specimens each imaged at seven time points (Day 0, Day 3, Week 1, Week

337 4, Week 8, Week 20) with available brain masks. In-plane resolution is 0*.*1 *×* 0*.*1 *mm*2 with

338 a slice thickness of 0*.*5 *mm*. Since the training data is isotropic and data augmentation

339 includes downsampling in the canonical directions, each of the two networks learns mouse

340 brain-specific interpolation such that one can perform prediction on thick-sliced images, as, for example, in these evaluation data, and return isotropic probability and thickness maps (a choice avail-

342 able to the user). Figure [6](#_bookmark5) summarizes the results of the evaluation and comparison between

343 isotropic and anisotropic cortical measurements in male and female specimens.

344 **3 Discussion**

345 The ANTsX ecosystem is a powerful framework that has demonstrated applicability to mul-

346 tiple species and organ systems, including the mouse brain. This is further evidenced by

347 the many other software packages that use various ANTsX components in their own mouse-

348 specific workflows. In and of itself, the extensive functionality of ANTsX makes it possible

349 to create complete processing pipelines without requiring the integration of multiple pack-

350 ages. These open-source components not only perform well but are available across multiple

351 platforms which facilitates the construction of tailored pipelines for individual study solu-

352 tions. These components are also supported by years of development not only by the ANTsX

353 development team but by the larger ITK community.

354 In the case of the development of the DevCCF, ANTsX was crucial in providing necessary

355 functionality for yielding high quality output. For the generation of the individual develop-

356 mental stage multi-modal, symmetric templates, ANTsX is unique amongst image analysis

357 software packages in providing existing solutions for template generation which have been

358 thoroughly vetted, including being used in several studies over the years, and which continue

359 to be under active refinement. At its core, computationally efficient and quality template

360 generation requires the use of precision pairwise image mapping functionality which, his-

361 torically, is at the origins of the ANTsX ecosystem. Moreover, these mapping capabilities extend

362 beyond template generation to the mapping of other image data (e.g., gene expression maps)

363 to a selected template for providing further insight into the mouse brain.

364 With respect to the DevCCF, despite the significant expansion of available developmental

365 age templates beyond what existed previously, there are still temporal gaps in the DevCCF

366 which can be potentially sampled by future research efforts. However, pioneering work

367 involving time-varying diffeomorphic transformations allow us to continuously situate the

368 existing templates within a velocity flow model. This allows one to determine the diffeomor-

369 phic transformation from any one temporal location to any other temporal location within

370 the time span defined by the temporal limits of the DevCCF. This functionality is built on

371 multiple components from the Insight Segmentation and Registration Toolkit including the

372 B-spline scattered data approximation technique for field regularization and velocity field in-

373 tegration. This velocity field model permits intra-template comparison and the construction

374 of virtual templates where a template can be estimated at any continuous time point within

375 the temporal domain. This novel application can potentially enhance our understanding of

376 intermediate developmental stages.

377 We also presented a mouse brain pipeline for brain extraction, parcellation, and cortical

378 thickness using single-shot and two-shot learning with data augmentation. This approach

379 attempts to circumvent (or at least minimize) the typical requirement of large training

380 datasets as with the human ANTsX pipeline analog. However, even given our initial success

381 on independent data, we fully anticipate that refinements will be necessary. In fact, a current

382 parallel study with a separate collaborator using private data yielded three brain extraction

383 failures (out of 89 specimens). Given that the ANTsX toolkit is a dynamic effort undergoing

384 continual improvement, we manually correct cases that fail and use them for future training and

385 refinement of network weights as we have done for our human-based networks. Generally,

386 these approaches provide a way to bootstrap training data for manual refinement and future

387 generation of more accurate deep learning networks in the absence of corresponding non deep

388 learning-based tools.

389 **4 Methods**

390 The following methods are all available as part of the ANTsX ecosystem with analogous

391 elements existing in both ANTsR (ANTs in R) and ANTsPy (ANTs in Python) with and

392 ANTs/ITK C++ core. However, most of the development for the work described below was

393 performed using ANTsPy. For equivalent calls in ANTsR, please see the ANTsX tutorial at

394 <https://tinyurl.com/antsxtutorial>.

# 395 4.1 General ANTsX utilities

## 396 4.1.1 Preprocessing: bias field correction and denoising

397 Bias field correction and image denoising are standard preprocessing steps in improving over-

398 all image quality in mouse brain images. The bias field, a gradual spatial intensity variation

399 in images, can arise from various sources such as magnetic field inhomogeneity or acquisition

400 artifacts, leading to distortions that can compromise the quality of brain images. Correct-

401 ing for bias fields ensures a more uniform and consistent representation of brain structures,

402 enabling more accurate quantitative analysis. Additionally, brain images are often suscep-

403 tible to various forms of noise, which can obscure subtle features and affect the precision

404 of measurements. Denoising techniques help mitigate the impact of noise, enhancing the

405 signal-to-noise ratio and improving the overall image quality. The well-known N4 bias field

406 correction algorithm25 has its origins in the ANTs toolkit which was implemented and intro-

407 duced into the ITK toolkit, i.e. ants.n4\_bias\_field\_correction(...). Similarly, ANTsX

408 contains an implementation of a well-performing patch-based denoising technique47 and is

409 also available as an image filter to the ITK community, ants.denoise\_image(...).

## 410 4.1.2 Image registration

411 The ANTs registration toolkit is a complex framework permitting highly tailored solu-

412 tions to pairwise image registration scenarios.68 It includes innovative transformation mod-

413 els for biological modeling41,53 and has proven capable of excellent performance.42,69 Var-

414 ious parameter sets targeting specific applications have been packaged with the different

415 ANTsX platforms, specifically ANTs, ANTsPy, and ANTsR.26 In ANTsPy, the function

416 ants.registration(...) is used to register a pair of images or a pair of image sets where

417 type\_of\_transform is a user-specified option that invokes a specific parameter set. For ex-

418 ample type\_of\_transform='antsRegistrationSyNQuick[s]' encapsulates an oft-used pa-

419 rameter set for quick registration whereas type\_of\_transform='antsRegistrationSyN[s]'

420 is a more detailed alternative. Transforming images using the derived transforms is performed

421 via the ants.apply\_transforms(...) function.

422 Initially, linear optimization is initialized with center of (intensity) mass alignment typically

423 followed by optimization of both rigid and affine transforms using the mutual information

424 similarity metric. This is followed by diffeomorphic deformable alignment using symmetric

425 normalization (SyN) with Gaussian41 or B-spline regularization53 where the forward trans-

426 form is invertible and differentiable. The similarity metric employed at this latter stage

427 is typically either neighborhood cross-correlation or mutual information. Note that these

428 parameter sets are robust to input image type (i.e., LSFM, Nissl staining, and the various

429 MRI modalities) and are adaptable to mousing image geometry scaling. Further details can

430 be found in the various documentation sources for these ANTsX packages.

## 431 4.1.3 Template generation

432 ANTsX provides functionality for constructing population templates from a set (or multi-modal sets) of

433 input images as originally described45 and recently used to create the DevCCF templates.15

434 An initial template estimate is constructed from an existing subject image or a voxelwise

435 average derived from a rigid pre-alignment of the image population. Pairwise registration

436 between each subject and the current template estimate is performed using the Symmetric

437 Normalization (SyN) algorithm.41 The template estimate is updated by warping all subjects

438 to the space of the template, performing a voxelwise average, and then performing a “shape

439 update” of this latter image by warping it by the average inverse deformation, thus yielding

440 a mean image of the population in terms of both intensity and shape. The corresponding

441 ANTsPy function is ants.build\_template(...).

## 442 4.1.4 Visualization

443 To complement the well-known visualization capabilities of R and Python, e.g., ggplot2

444 and matplotlib, respectively, image-specific visualization capabilities are available in the

445 ants.plot(...)function (Python). These are capable of illustrating multiple slices in different

446 orientations with both other image overlays as well as label images.

# 447 4.2 Mapping fMOST data to AllenCCFv3

## 448 4.2.1 Preprocessing

449 • *Downsampling*. The first challenge when mapping fMOST images into the AllenCCFv3

450 is addressing the resolution scale of the data. Native fMOST data from an individual

451 specimen can range in the order of terabytes, which leads to two main problems. First,

452 volumetric registration methods (particularly those estimating local deformation) have

453 high computational complexity and typically cannot operate on such high-resolution

454 data under reasonable memory and runtime constraints. Second, the resolution of

455 the AllenCCFv3 atlas is much lower than the fMOST data, thus the mapping process

456 will cause much of the high-resolution information in the fMOST images to be lost

457 regardless. Thus, we perform a cubic B-spline downsampling of the fMOST data to

458 reduce the resolution of each image ~~to 25~~ *~~µm~~* ~~isotropic~~ to match the isotropic 25 *µm voxel resolution of the* AllenCCFv3

459 intensity atlas using ants.resample\_image(...). An important detail to note is that

460 while the fMOST images and atlas are downsampled, the mapping learned during the

461 registration is assumed to be continuous. Thus, after establishing the mapping to

462 the AllenCCFv3, we can interpolate the learned mapping and apply it directly to the high-

463 resolution native data to transform any spatially aligned data (such as the

464 single-cell neuron reconstructions) into the AllenCCFv3.

465 • *Stripe artifact removal*. Repetitive pattern artifacts are a common challenge in fMOST

466 imaging where inhomogeneity during the cutting and imaging of different sections can

467 leave stripes of hyper- and hypo-intensity across the image. These stripe artifacts

468 can be latched onto by the registration algorithm as unintended features that are

469 then misregistered to non-analogous structures in the AllenCCFv3. We address these

470 artifacts by fitting a 3D bandstop (notch) filter to target the frequency of the stripe

471 patterns and removing them prior to the image registration.

472 • *Inhomogeneity correction*. Regional intensity inhomogeneity can also occur within

473 and between sections in fMOST imaging due to staining or lighting irregularity dur-

474 ing acquisition. Similar to stripe artifacts, intensity gradients due to inhomogeneity

475 can be misconstrued as features during the mapping and result in matching of non-

476 corresponding structures. Our pipeline addresses these intensity inhomogeneities using

477 N4 bias field correction,25 ants.n4\_bias\_field\_correction(...).

## 478 4.2.2 ~~Steps for~~ spatial normalization to AllenCCFv3

479 1. *Average fMOST atlas as an intermediate target*. Due to the preparation of the mouse

480 brain for fMOST imaging, the resulting structure in the mouse brain has several large

481 morphological deviations from the AllenCCFv3 atlas. Most notable of these is an

482 enlargement of the ventricles, and compression of cortical structures. In addition,

483 there is poor intensity correspondence for the same anatomic features due to intensity

484 dissimilarity between imaging modalities. We have found that standard intensity-based

485 registration is insufficient to capture the significant deformations required to map these

486 structures correctly into the AllenCCFv3. We address this challenge in ANTsX by using

487 explicitly corresponding parcellations of the brain, ventricles and surrounding struc-

488 tures to directly recover these large morphological differences. However, generating these

489 parcellations for each individual mouse brain is a labor-intensive task. Our solution

490 is to create an average atlas whose mapping to AllenCCFv3 encapsulates these large morphological differences to

491 serve as an intermediate registration point. This has the advantage of only needing to

492 generate one set of corresponding annotations which is used to register between the

493 two atlas spaces. New images are first aligned to the fMOST average atlas, which

494 shares common intensity and morphological features and thus can be achieved through

495 standard intensity-based registration.

496 2. *Average fMOST atlas construction*. An intensity and shape-based contralaterally sym-

497 metric average of the fMOST image data is constructed from 30 images and their

498 contralateral counterpart. We ran three iterations of the atlas construction using the

499 default settings. Additional iterations (up to six) were evaluated and showed minimal

500 changes to the final atlas construction, suggesting a convergence of the algorithm.

501 3. *fMOST atlas to AllenCCFv3 alignment*. Alignment between the fMOST average atlas

502 and AllenCCFv3 was performed using a one-time annotation-driven approach. Label-

503 to-label registration is used to align 7 corresponding annotations in both atlases in

504 the following: 1) brain mask/ventricles, 2) caudate/putamen, 3) fimbria, 4) posterior

505 choroid plexus, 5) optic chiasm, 6) anterior choroid plexus, and 7) habenular com-

506 missure. The alignments were performed sequentially, with the largest, most relevant

507 structures being aligned first using coarse registration parameters, followed by other

508 structures using finer parameters. This coarse-to-fine approach allows us to address large morpholog-

509 ical differences (such as brain shape and ventricle expansion) at the start of registration

510 and then progressively refine the mapping using the smaller structures. The overall ordering of these

511 structures was determined manually by an expert anatomist, where anatomical mis-

512 registration after each step of the registration was evaluated and used to determine

513 which structure should be used in the subsequent iteration to best improve the align-

514 ment. The transformation from this one-time expert-guided alignment is preserved and used as the

515 canonical fMOST atlas to AllenCCFv3 mapping in the pipeline.

516 4. *Alignment of individual fMOST mouse brains*. The canonical transformation between

517 the fMOST atlas and AllenCCFv3 greatly simplifies the registration of new individual

518 fMOST mouse brains into the AllenCCFv3. Each new image is first registered into the

519 fMOST average atlas, which shares intensity, modality, and morphological characteris-

520 tics. This allows us to leverage standard, intensity-based registration functionality available68 in ANTsX to perform

521 this alignment. Transformations are then concatenated to the original fMOST image to

522 move it into the AllenCCFv3 space using ants.apply\_transforms(...).

523 5. *Transformation of single cell neurons*. A key feature of fMOST imaging is the ability

524 to reconstruct and examine whole-brain single neuron projections63. Spatial mapping

525 of these neurons from individual brains into the AllenCCFv3 allows investigators to

526 study different neuron types within the same space and characterize their morphology

527 with respect to their transcriptomics. Mappings found between the fMOST image and

528 the AllenCCFv3 using our pipeline can be applied in this way to fMOST neuron reconstruction

529 data.

# 530 4.3 Mapping MERFISH data to AllenCCFv3

## 531 4.3.1 Preprocessing

532 • *Initial volume reconstruction*—Alignment of MERFISH data into a 3D atlas space requires

533 an estimation of anatomical structure within the data. For each section, this anatomic

534 reference image was created by aggregating the number of detected genetic markers

535 (across all probes) within each pixel of a 10 *×* 10 *µm*2 grid to match the resolution

536 of the 10 *µm* AllenCCFv3 atlas. These reference image sections are then coarsely re-

537 oriented and aligned across sections using manual annotations of the most dorsal and

538 ventral points of the midline. The procedure produces an anatomic image stack that

539 serves as an initialization for further global mappings into the AllenCCFv3.

540 • *Anatomical correspondence labeling*—Mapping the MERFISH data into the AllenC-

541 CFv3 requires us to establish correspondence between the anatomy depicted in the MERFISH and AllenCCFv3

542 data. Intensity-based features in MERFISH data are not sufficiently apparent to es-

543 tablish this correspondence, so we generate instead corresponding anatomical

544 labelings of both images with which to drive registration. These labels are already available as part of the AllenCCFv3;

545 thus, the main challenge is deriving analogous labels from the spatial transcriptomic

546 maps of the MERFISH data. Toward this end, we assigned each cell from

547 the scRNA-seq dataset to one of the following major regions: cerebellum, CTXsp, hindbrain,

548 HPF, hypothalamus, isocortex, LSX, midbrain, OLF, PAL, sAMY, STRd, STRv, tha-

549 lamus and hindbrain. A label map of each section was generated for each region by

550 aggregating the cells assigned to that region within a 10 *×* 10 *µm*2 grid. The same

551 approach was used to generate more fine grained region specific landmarks (i.e. cortical

552 layers, habenula, IC). Unlike the first set of labels which cover the entirety of the section

553 these latter regions are highly specific to certain parts of the section. Once cells in the MER-

554 FISH data are labeled, morphological dilation is used to provide regional labels without any gaps for

555 alignment into the AllenCCFv3.

556 • *Section matching*—Since the MERFISH data is acquired as sections, its 3D orientation

557 may not be fully accounted for during the volume reconstruction step, due to the particular cutting

558 angle. This can lead to obliqueness artifacts in the section where certain structures can

559 appear to be larger or smaller, or missing outright from the section. To address this, we

560 first use a global alignment to match the orientations of the MERFISH sections to the

561 atlas space. In our pipeline, this section matching is performed in the reverse direction

562 by performing a global affine transformation of the AllenCCFv3 into the MERFISH

563 data space, and then resampling digital sections from the AllenCCFv3 to match each

564 MERFISH section. This approach limits the overall transformation and thus resampling that is applied to

565 the MERFISH data, and, since the AllenCCFv3 is densely sampled, it also reduces in-

566 plane artifacts that result from missing sections or undefined spacing in the MERFISH

567 data.

## 568 4.3.2 2.5D deformable, landmark-driven alignment to AllenCCFv3

569 After global alignment of the AllenCCFv3 into the MERFISH dataset, 2D per-section de-

570 formable refinements are used to address local differences between the MERFISH sections

571 and the resampled AllenCCFv3 sections. Nine registrations were performed in sequence us-

572 ing a single label at each iteration in the following order: 1) brain mask, 2) isocortex (layer

573 2+3), 3) isocortex (layer 5), 4) isocortex (layer 6), 5) striatum, 6) medial habenula, 7) lateral

574 habenula, 8) thalamus, and 9) hippocampus. This ordering was determined empirically by an

575 expert anatomist who prioritized which structure to use in each iteration by evaluating the

576 anatomical alignment from the previous iteration. Global and local mappings are then all

577 concatenated (with appropriate inversions) to create the final mapping between the MER-

578 FISH data and AllenCCFv3 (Figure 7). This mapping is then used to provide a point-to-point

579 correspondence between the original MERFISH coordinate space and the AllenCCFv3 space,

580 thus allowing mapping of individual genes and cell types located in the MERFISH data to

581 be directly mapped into the AllenCCFv3.

# 582 4.4 DevCCF velocity flow transformation model

583 Given multiple, linearly or non-linearly ordered point sets where individual points across the sets are

584 in one-to-one correspondence, we developed an approach for generating a velocity flow trans-

585 formation model to describe a time-varying diffeomorphic mapping as a variant of the inexact

586 landmark matching solution. Integration of the resulting velocity field can then be used to

587 describe the displacement between any two time points within this time-parameterized do-

588 main. Regularization of the sparse correspondence between point sets is performed using a

589 generalized B-spline scattered data approximation technique,67 also created by the ANTsX

590 developers and contributed to ITK.

## 591 4.4.1 Velocity field optimization

592 To apply this methodology to the developmental templates,15 we coalesced the manual

593 annotations of the developmental templates into 26 common anatomical regions (13 per

594 hemisphere). We then used these regions to generate invertible transformations between

595 successive time points. Specifically, each label was used to create a pair of single region

596 images resulting in 26 pairs of “source” and “target” images. The multiple image pairs

597 were used to iteratively estimate a diffeomorphic pairwise transform. Given the seven at-

598 lases E11.5, E13.5, E15.5, E18.5, P4, P14, and P56, this resulted in 6 sets of transforms

599 between successive time points. Given the relative sizes between atlases (ranging from X to X), on the order of

600 106 points were randomly sampled labelwise in the P56 template space and propagated

601 to each successive atlas providing the point sets for constructing the velocity flow model.

602 Approximately 125 iterations resulted in a steady convergence based on the average Eu-

603 clidean norm between transformed point sets. Ten integration points were used and point

604 sets were distributed along the temporal dimension using a log transform for a more evenly

605 spaced sampling. For additional information, see the help menu for the ANTsPy function

606 ants.fit\_time\_varying\_transform\_to\_point\_sets(...).

# 607 4.5 ANTsXNet mouse brain applications

## 608 4.5.1 General notes regarding deep learning training

609 All network-based approaches described below were implemented and organized in the

610 ANTsXNet libraries comprising Python (ANTsPyNet) and R (ANTsRNet) analogs using the

611 Keras/Tensorflow libraries available as open source in ANTsX GitHub repositories. For the

612 various applications, both libraries share the identically trained weights for mutual reproducibility.

613 For all GPU training, we used Python scripts for creating custom batch generators. As

614 such batch generators tend to be application-specific, we store them in a separate GitHub

615 repository for public availability (<https://github.com/ntustison/ANTsXNetTraining>). In

616 terms of GPU hardware, all training was performed on a DGX (GPUs: 4X Tesla V100, system

617 memory: 256 GB LRDIMM DDR4).

618 Data augmentation is crucial for generalizability and accuracy of the trained networks.

619 Intensity-based data augmentation consisted of randomly added noise (i.e., Gaussian, shot,

620 salt-and-pepper), simulated bias fields based on N4 bias field modeling, and histogram warp-

621 ing for mimicking well-known MRI intensity nonlinearities.26,70 These augmentation tech-

622 niques are available in ANTsXNet (only ANTsPyNet versions are listed with ANTsRNet

623 versions available) and include:

624 • image noise: ants.add\_noise\_to\_image(...),

625 • simulated bias field: antspynet.simulate\_bias\_field(...), and

626 • nonlinear intensity warping: antspynet.histogram\_warp\_image\_intensities(...).

627 Shape-based data augmentation used both random linear and nonlinear deformations in

628 addition to anisotropic resampling in the three canonical orientations to mimic frequently

629 used acquisition protocols for mouse brains:

630 • random spatial warping: antspynet.randomly\_transform\_image\_data(...) and

631 • anisotropic resampling: ants.resample\_image(...).

## 632 4.5.2 Brain extraction

633 Similar to human neuroimage processing, brain extraction is a crucial preprocessing step

634 for accurate brain mapping. Within ANTsXNet, we have created several deep learning

635 networks for brain extraction for the following MRI-based modalities: T1, FLAIR, and fractional

636 anisotropy. ~~Similarly,~~ For the developmental brain atlas work15 we developed similar func-

637 tionality for mouse brains of different modalities and developmental age. All networks use

638 a conventional U-net architecture.71 Whereas T2-weighted brain extraction is volumetric-

639 based for both isotropic and anisotropic data, coronal and sagittal networks are available for

640 both E13.5 and E15.5 data because XXXX. In ANTsPyNet, this functionality is available in the program

641 antspynet.mouse\_brain\_extraction(...).

642 For the two-shot T2-weighted brain extraction network, two brain templates were generated

643 along with their masks. One of the templates was generated from orthogonal multi-

644 plane, high resolution data58 synthesized isotropic volumetric data using the B-spline fitting

645 algorithm.67 This algorithm is encapsulated in ants.fit\_bspline\_object\_to\_scattered\_data(...),

646 where the input is the set of voxel intensity values and associated physical location. Since

647 each point can be assigned a confidence weight, we use the the normalized gradient value

648 to more heavily weight edge regions. Although both template/mask pairs are available

649 in the GitHub repository associated with this work, the synthesized volumetric B-spline

650 T2-weighted pair is available within ANTsXNet through the calls:

651 • template: antspynet.get\_antsxnet\_data("bsplineT2MouseTemplate") and

652 • mask: antspynet.get\_antsxnet\_data("bsplineT2MouseTemplateBrainMask").

## 653 4.5.3 Brain parcellation

654 The T2-weighted brain parcellation network is also based on a 3-D U-net architecture and the

655 T2-w DevCCF P56 template component with extensive data augmentation, as described pre-

656 viously. Intensity differences between the template and any brain extracted input image are

657 minimized through the use of the rank intensity transform (ants.rank\_intensity(...)).

658 Shape differences are reduced by the additional preprocessing step of warping the brain ex-

659 tracted input image to the template. Additional input channels include the prior probability

660 images created from the template parcellation. These images are also available through the

661 ANTsXNet interface:

662 • template: antspynet.get\_antsxnet\_data("DevCCF\_P56\_MRI-T2\_50um") and

663 • parcellation: antspynet.get\_antsxnet\_data("DevCCF\_P56\_MRI-T2\_50um\_BrainParcellationNic

664 **Data availability.** All data and software used in this work are publicly available. The

665 DevCCF atlas is available at <https://kimlab.io/brain-map/DevCCF/>. ANTsPy, ANTsR,

666 ANTsPyNet, and ANTsRNet are available through GitHub at the ANTsX Ecosystem ([https:](https://github.com/ANTsX)

667 [//github.com/ANTsX](https://github.com/ANTsX)). Training scripts for all deep learning functionality in ANTsXNet can

668 also be found on GitHub (<https://github.com/ntustison/ANTsXNetTraining>). A GitHub

669 repository specifically pertaining to the AllenCCFv3 mappings is available at [https://github.](https://github.com/dontminchenit/CCFAlignmentToolkit)

670 [com/dontminchenit/CCFAlignmentToolkit](https://github.com/dontminchenit/CCFAlignmentToolkit). For the other two contributions contained in

671 this work, the longitudinal DevCCF mapping and mouse cortical thickness pipeline, we refer

672 the interested reader to <https://github.com/ntustison/ANTsXMouseBrainMapping>.

673 **References**

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

1. Keller, P. J. & Ahrens, M. B. Visualizing whole-brain activity and development at the single-cell level using light-sheet microscopy. *Neuron* **85**, 462–83 (2015).
2. La Manno, G. *et al.* Molecular architecture of the developing mouse brain. *Nature*

**596**, 92–96 (2021).

1. Wen, L. *et al.* Single-cell technologies: From research to application. *Innovation (Camb)* **3**, 100342 (2022).
2. Oh, S. W. *et al.* A mesoscale connectome of the mouse brain. *Nature* **508**, 207–14 (2014).
3. Gong, H. *et al.* Continuously tracing brain-wide long-distance axonal projections in mice at a one-micron voxel resolution. *Neuroimage* **74**, 87–98 (2013).
4. Li, A. *et al.* Micro-optical sectioning tomography to obtain a high-resolution atlas of the mouse brain. *Science* **330**, 1404–8 (2010).
5. Ueda, H. R. *et al.* Tissue clearing and its applications in neuroscience. *Nat Rev Neurosci* **21**, 61–79 (2020).
6. Ståhl, P. L. *et al.* Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science* **353**, 78–82 (2016).
7. Burgess, D. J. Spatial transcriptomics coming of age. *Nat Rev Genet* **20**, 317 (2019).
8. MacKenzie-Graham, A. *et al.* A multimodal, multidimensional atlas of the C57BL/6J mouse brain. *J Anat* **204**, 93–102 (2004).
9. Mackenzie-Graham, A. J. *et al.* Multimodal, multidimensional models of mouse brain.

*Epilepsia* **48 Suppl 4**, 75–81 (2007).

1. Dong, H. W. *Allen reference atlas. A digital color brain atlas of the C57BL/6J male mouse*. (John Wiley; Sons, 2008).

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

1. Wang, Q. *et al.* The allen mouse brain common coordinate framework: A 3D reference atlas. *Cell* **181**, 936–953.e20 (2020).
2. Johnson, G. A. *et al.* Waxholm space: An image-based reference for coordinating mouse brain research. *Neuroimage* **53**, 365–72 (2010).
3. Kronman, F. A. *et al.* Developmental mouse brain common coordinate framework.

*bioRxiv* (2023) doi:[10.1101/2023.09.14.557789](https://doi.org/10.1101/2023.09.14.557789).

1. Oguz, I., Zhang, H., Rumple, A. & Sonka, M. RATS: Rapid automatic tissue segmen- tation in rodent brain MRI. *J Neurosci Methods* **221**, 175–82 (2014).
2. Sawiak, S. J., Picq, J.-L. & Dhenain, M. Voxel-based morphometry analyses of in vivo MRI in the aging mouse lemur primate. *Front Aging Neurosci* **6**, 82 (2014).
3. Ashburner, J. SPM: A history. *Neuroimage* **62**, 791–800 (2012).
4. Modat, M. *et al.* Fast free-form deformation using graphics processing units. *Comput Methods Programs Biomed* **98**, 278–84 (2010).
5. Tyson, A. L. *et al.* Accurate determination of marker location within whole-brain microscopy images. *Sci Rep* **12**, 867 (2022).
6. Pallast, N. *et al.* Processing pipeline for atlas-based imaging data analysis of struc- tural and functional mouse brain MRI (AIDAmri). *Front Neuroinform* **13**, 42 (2019).
7. Jenkinson, M., Beckmann, C. F., Behrens, T. E. J., Woolrich, M. W. & Smith, S. M. FSL. *Neuroimage* **62**, 782–90 (2012).
8. Yeh, F.-C., Wedeen, V. J. & Tseng, W.-Y. I. Generalized q-sampling imaging. *IEEE Trans Med Imaging* **29**, 1626–35 (2010).
9. Jorge Cardoso, M. *et al.* STEPS: Similarity and truth estimation for propagated segmentations and its application to hippocampal segmentation and brain parcelation. *Med Image Anal* **17**, 671–84 (2013).

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

1. Tustison, N. J. *et al.* N4ITK: Improved N3 bias correction. *IEEE Trans Med Imaging*

**29**, 1310–20 (2010).

1. Tustison, N. J. *et al.* The ANTsX ecosystem for quantitative biological and medical imaging. *Sci Rep* **11**, 9068 (2021).
2. Goubran, M. *et al.* Multimodal image registration and connectivity analysis for inte- gration of connectomic data from microscopy to MRI. *Nat Commun* **10**, 5504 (2019).
3. Celestine, M., Nadkarni, N. A., Garin, C. M., Bougacha, S. & Dhenain, M. Sammba- MRI: A library for processing SmAll-MaMmal BrAin MRI data in python. *Front Neuroinform* **14**, 24 (2020).
4. Ioanas, H.-I., Marks, M., Zerbi, V., Yanik, M. F. & Rudin, M. An optimized regis- tration workflow and standard geometric space for small animal brain imaging. *Neu- roimage* **241**, 118386 (2021).
5. Cox, R. W. AFNI: What a long strange trip it’s been. *Neuroimage* **62**, 743–7 (2012).
6. Anderson, R. J. *et al.* Small animal multivariate brain analysis (SAMBA) - a high throughput pipeline with a validation framework. *Neuroinformatics* **17**, 451–472 (2019).
7. Ni, H. *et al.* A robust image registration interface for large volume brain atlas. *Sci Rep* **10**, 2139 (2020).
8. Jin, M. *et al.* SMART: An open-source extension of WholeBrain for intact mouse brain registration and segmentation. *eNeuro* **9**, (2022).
9. Fürth, D. *et al.* An interactive framework for whole-brain maps at cellular resolution.

*Nat Neurosci* **21**, 139–149 (2018).

1. Negwer, M. *et al.* FriendlyClearMap: An optimized toolkit for mouse brain mapping and analysis. *Gigascience* **12**, (2022).

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

1. Klein, S., Staring, M., Murphy, K., Viergever, M. A. & Pluim, J. P. W. Elastix: A toolbox for intensity-based medical image registration. *IEEE Trans Med Imaging* **29**, 196–205 (2010).
2. Carey, H. *et al.* DeepSlice: Rapid fully automatic registration of mouse brain imaging to a volumetric atlas. *Nat Commun* **14**, 5884 (2023).
3. Bajcsy, R. & Broit, C. Matching of deformed images. in *Sixth International Confer- ence on Pattern Recognition (ICPR’82)* 351–353 (1982).
4. Bajcsy, R. & Kovacic, S. Multiresolution elastic matching. *Computer Vision, Graph- ics, and Image Processing* **46**, 1–21 (1989).
5. Gee, J., Sundaram, T., Hasegawa, I., Uematsu, H. & Hatabu, H. Characterization of regional pulmonary mechanics from serial magnetic resonance imaging data. *Acad Radiol* **10**, 1147–52 (2003).
6. Avants, B. B., Epstein, C. L., Grossman, M. & Gee, J. C. Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. *Med Image Anal* **12**, 26–41 (2008).
7. Klein, A. *et al.* Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. *Neuroimage* **46**, 786–802 (2009).
8. Murphy, K. *et al.* Evaluation of registration methods on thoracic CT: The EMPIRE10 challenge. *IEEE Trans Med Imaging* **30**, 1901–20 (2011).
9. Baheti, B. *et al.* The brain tumor sequence registration challenge: Establishing corre- spondence between pre-operative and follow-up MRI scans of diffuse glioma patients. (2021).
10. Avants, B. B. *et al.* The optimal template effect in hippocampus studies of diseased populations. *Neuroimage* **49**, 2457–66 (2010).

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

1. Avants, B. B., Tustison, N. J., Wu, J., Cook, P. A. & Gee, J. C. An open source multivariate framework for n-tissue segmentation with evaluation on public data. *Neuroinformatics* **9**, 381–400 (2011).
2. Manjón, J. V., Coupé, P., Martí-Bonmatí, L., Collins, D. L. & Robles, M. Adaptive non-local means denoising of MR images with spatially varying noise levels. *J Magn Reson Imaging* **31**, 192–203 (2010).
3. Wang, H. *et al.* Multi-atlas segmentation with joint label fusion. *IEEE Trans Pattern Anal Mach Intell* **35**, 611–23 (2013).
4. Tustison, N. J. *et al.* Optimal symmetric multimodal templates and concatenated random forests for supervised brain tumor segmentation (simplified) with *ANTsR*. *Neuroinformatics* (2014) doi:[10.1007/s12021-014-9245-2](https://doi.org/10.1007/s12021-014-9245-2).
5. Tustison, N. J., Yang, Y. & Salerno, M. Advanced normalization tools for cardiac mo- tion correction. in *Statistical atlases and computational models of the heart - imaging and modelling challenges* (eds. Camara, O. et al.) vol. 8896 3–12 (Springer Interna- tional Publishing, 2015).
6. McCormick, M., Liu, X., Jomier, J., Marion, C. & Ibanez, L. ITK: Enabling repro- ducible research and open science. *Front Neuroinform* **8**, 13 (2014).
7. Beg, M. F., Miller, M. I., Trouvé, A. & Younes, L. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision* **61**, 139–157 (2005).
8. Tustison, N. J. & Avants, B. B. Explicit B-spline regularization in diffeomorphic image registration. *Front Neuroinform* **7**, 39 (2013).
9. Das, S. R., Avants, B. B., Grossman, M. & Gee, J. C. Registration based cortical thickness measurement. *Neuroimage* **45**, 867–79 (2009).
10. Tustison, N. J. *et al.* Large-scale evaluation of ANTs and FreeSurfer cortical thickness measurements. *Neuroimage* **99**, 166–79 (2014).

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

1. Tustison, N. J. *et al.* Longitudinal mapping of cortical thickness measurements: An Alzheimer’s Disease Neuroimaging Initiative-based evaluation study. *J Alzheimers Dis* (2019) doi:[10.3233/JAD-190283](https://doi.org/10.3233/JAD-190283).
2. Hsu, L.-M. *et al.* CAMRI mouse brain MRI data.
3. Reshetnikov, V. *et al.* High-resolution MRI data of brain C57BL/6 and BTBR mice in three different anatomical views.
4. Rahman, N., Xu, K., Budde, M. D., Brown, A. & Baron, C. A. A longitudinal microstructural MRI dataset in healthy C57Bl/6 mice at 9.4 tesla. *Sci Data* **10**, 94 (2023).
5. Gong, H. *et al.* High-throughput dual-colour precision imaging for brain-wide connec- tome with cytoarchitectonic landmarks at the cellular level. *Nat Commun* **7**, 12142 (2016).
6. Wang, J. *et al.* Divergent projection patterns revealed by reconstruction of individual neurons in orbitofrontal cortex. *Neurosci Bull* **37**, 461–477 (2021).
7. Rotolo, T., Smallwood, P. M., Williams, J. & Nathans, J. Genetically-directed, cell type-specific sparse labeling for the analysis of neuronal morphology. *PLoS One* **3**, e4099 (2008).
8. Peng, H. *et al.* Morphological diversity of single neurons in molecularly defined cell types. *Nature* **598**, 174–181 (2021).
9. Yao, Z. *et al.* A high-resolution transcriptomic and spatial atlas of cell types in the whole mouse brain. *Nature* **624**, 317–332 (2023).
10. Liu, J. *et al.* Concordance of MERFISH spatial transcriptomics with bulk and single- cell RNA sequencing. *Life Sci Alliance* **6**, (2023).
11. Stringer, C., Wang, T., Michaelos, M. & Pachitariu, M. Cellpose: A generalist algo- rithm for cellular segmentation. *Nat Methods* **18**, 100–106 (2021).

806

807

808

809

810

811

812

813

814

815

1. Tustison, N. J. & Amini, A. A. Biventricular myocardial strains via nonrigid registra- tion of anatomical NURBS model [corrected]. *IEEE Trans Med Imaging* **25**, 94–112 (2006).
2. Avants, B. B. *et al.* The Insight ToolKit image registration framework. *Front Neu- roinform* **8**, 44 (2014).
3. Avants, B. B. *et al.* A reproducible evaluation of ANTs similarity metric performance in brain image registration. *Neuroimage* **54**, 2033–44 (2011).
4. Nyúl, L. G., Udupa, J. K. & Zhang, X. New variants of a method of MRI scale standardization. *IEEE Trans Med Imaging* **19**, 143–50 (2000).
5. Falk, T. *et al.* U-net: Deep learning for cell counting, detection, and morphometry.

*Nat Methods* **16**, 67–70 (2019).