

## Structural bioinformatics

# Finding antibodies in cryo-EM maps with CrAI

Vincent Mallet<sup>1,\*</sup>, Chiara Rapisarda<sup>2,\*</sup>, Hervé Minoux<sup>2</sup>, Maks Ovsjanikov<sup>1</sup>

<sup>1</sup>LIX, Ecole Polytechnique, IPP Paris, Palaiseau, 91120, France

<sup>2</sup>Integrated Drug Discovery, Structural Biology and Biophysics, Sanofi, Vitry-sur-Seine, 94400, France

\*Corresponding authors. Vincent Mallet, CBIO, 60 boulevard saint Michel, 75006, Paris, France. E-mail: vincent.mallet@minesparis.psl.eu; Chiara Rapisarda, Sanofi, 1 impasse des Ateliers, Vitry-sur-Seine, 94400, France. E-mail: chiara.rapisarda@sanofi.com.

Associate Editor: Lenore Cowen

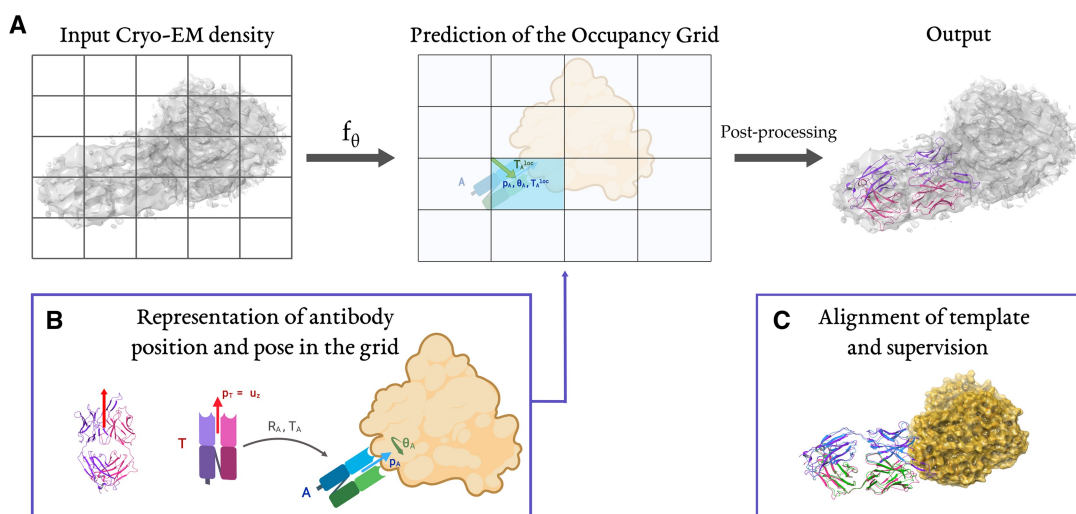
## Abstract

**Motivation:** Therapeutic antibodies have emerged as a prominent class of new drugs due to their high specificity and their ability to bind to several protein targets. Once an initial antibody has been identified, its design and characteristics are refined using structural information, when it is available. Cryo-EM is currently the most effective method to obtain 3D structures. It relies on well-established methods to process raw data into a 3D map, which may, however, be noisy and contain artifacts. To fully interpret these maps the number, position, and structure of antibodies and other proteins present must be determined. Unfortunately, existing automated methods addressing this step have limited accuracy, require additional inputs and high-resolution maps, and exhibit long running times.

**Results:** We propose the first fully automatic and efficient method dedicated to finding antibodies in cryo-EM maps: CrAI. This machine learning approach leverages the conserved structure of antibodies and a dedicated novel database that we built to solve this problem. Running a prediction takes only a few seconds, instead of hours, and requires nothing but the cryo-EM map, seamlessly integrating within automated analysis pipelines. Our method can find the location and pose of both Fabs and VHHs at resolutions up to 10 Å and is significantly more reliable than existing approaches.

**Availability and implementation:** We make our method available both in open source [github.com/Sanofi-Public/crai](https://github.com/Sanofi-Public/crai) and as a ChimeraX bundle (crai).

## Graphical abstract



## 1 Introduction

Since the first monoclonal antibody entered the clinic in 1986 (Emmons and Hunsicker 1987), antibody-based therapeutics have made considerable progress. With over a hundred compounds approved and forty just in the last three years

(Kaplon *et al.* 2023, TheAntibodySociety 2023), the use of antibodies currently appears as one of the most promising approaches for designing new treatments for patients. Antibody-based therapeutics rely on the identification of antibodies that can bind to a target molecule (antigen) with

Received: 6 August 2024; Revised: 31 March 2025; Editorial Decision: 2 April 2025; Accepted: 7 April 2025

© The Author(s) 2025. Published by Oxford University Press.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

high specificity through their tips called the Complementarity Determining Regions (CDRs). Monoclonal antibodies (mAb) are the most widely used antibodies. They are composed of two domains called the Antigen Binding Fragment (Fab) and a central stalk that binds to the Fc receptor. More recently, antibody fragments consisting of a single variable antibody domain, engineered from heavy chain antibodies generated by camelids (Arbabi Ghahroudi *et al.* 1997) (nAbs or VHHs) have attracted interest as an alternative to mAbs (Duggan 2018, Jovčevska and Muyldermans 2020). Whether for mAbs or VHHs, initial hits are typically found using immunization (Köhler and Milstein 1975, Jones *et al.* 1986) or phage display (Smith 1985, McCafferty *et al.* 1990). Before entering clinical studies, initial hits need to be optimized with regard to several properties including their efficacy, manufacturability and safety. Among those properties, specific binding to the antigen is a critical objective scrutinized from the early phases of the process until drug candidate selection. Binding optimization relies on obtaining the structure of the initial hit since the knowledge of the atomic coordinates at the contact points between the Ab and its target improves the understanding of its mode of action and guides the optimization of the binding affinity (Chiu *et al.* 2019).

Cryogenic Electron Microscopy (cryo-EM) has become the most common way to experimentally obtain protein structures of therapeutic antibodies bound to their target. In cryo-EM, the target protein is embedded in ice and exposed to an electron beam, resulting in raw noisy images of individual particles. These raw 2D images are then aligned and transformed into a 3D Coulomb potential map (Scheres 2012, Punjani *et al.* 2017) from which atomic coordinates are inferred. Recent advances in data collection hardware and software, along with improved data processing pipelines, have increased data output. As a result, more academic labs and global pharmaceutical industries have adopted the technology (Peplow 2017). Recently, a new data collection workflow has been shown to produce 3–4 Å structures of a pharmaceutically relevant target protein with 1 h of instrument time, thus allowing the theoretical resolution of 24 structures a day (Cushing *et al.* 2023). However, this raw data needs to be processed from micrographs to molecular structures. While data collection is rapid, current data *processing pipelines* rely on significant manual intervention for simple tasks and decisions, and ultimately take days and weeks to complete.

The recent rise of artificial intelligence-based methods for protein structure prediction holds promise but cannot yet accurately model the interaction between antibodies and their epitope. Current methods fail to accurately model the diverse CDR loops that are critical for antigen binding (Hummer *et al.* 2022) calling for significantly more structural data. Unfortunately, the reliance of existing methods on manual intervention significantly hinders cryo-EM map analysis at scale, required for such understanding.

As automation has allowed X-ray crystallography to become a key technique in the structure-based drug discovery pipeline (Blundell and Patel 2004), so it should for cryo-EM, making it cheaper and faster, and freeing the time of researchers from button-pressing tasks, to structure interpretation and drug engineering. The process of automation is being accelerated by the application of machine learning to the different stages of the pipeline, from data acquisition (Bouvette *et al.* 2022, Fan *et al.* 2022), to preprocessing of micrographs (Sanchez-Garcia *et al.* 2020), particle picking (Wang *et al.*

2016, Bepler *et al.* 2019, Wagner *et al.* 2019, Zhang *et al.* 2019, Yao *et al.* 2020), 2D class selection (Kimanius *et al.* 2021), 3D heterogeneity deconvolution (Matsumoto *et al.* 2021, Zhong *et al.* 2021), and analysis (Pearce *et al.* 2017, Brzezinski *et al.* 2021, Karolczak *et al.* 2024).

Unfortunately, the last step of attribution of the map (fitting of atomic coordinates of a protein into the map), remains a tedious analysis bottleneck. It is still a largely manual process typically done using ChimeraX (Pettersen *et al.* 2004), followed by the local optimization of atomic coordinates using Coot (Emsley *et al.* 2010). Some techniques have been developed to help automate this process (Liebschner *et al.* 2019), although with limited accuracy. Furthermore, the methodological challenges associated with this problem have so far impeded the automation of this step using existing machine learning approaches. Specifically, the data is noisy and heterogeneous and the output is high dimensional, making off-the-shelf computer vision methods irrelevant. Tools based on machine learning to trace the sequence in the map were recently developed with good results (Pfab *et al.* 2021, Jamali *et al.* 2024, Wang *et al.* 2024). They are, however, limited to resolutions better than 4 Å and can exhibit prohibitive running times.

In the context of using cryo-EM for optimization of therapeutic Abs, we aim to address the problem of finding Abs (Fabs and VHHs) in cryo-EM maps. To achieve this, we propose CrAI, the first fully automatic and efficient machine learning based approach that is applicable at all resolutions better than 10 Å without any additional inputs beyond the map. To develop our solution, we introduce a customized deep learning technique, which takes into account and exploits the structural properties of this problem setting. In particular, we leverage the conserved structure of Abs as a prior information (Cohen *et al.* 2022) to formulate our problem as a special instance of 3D object detection (Qi *et al.* 2019, Zou *et al.* 2023). We gather a novel database of aligned Ab structures and Cryo-EM maps, and use it to train a model with a custom loss that involves optimal transport supervision. We test our tool on a set of 215 maps of various resolutions, containing 374 Fabs and 86 VHHs. We successfully find Abs in over 90% of systems, outperforming existing methods by a margin of 25%, while exhibiting *one thousand times* speedup. We make our tool available as a ChimeraX bundle to facilitate adoption.

## 2 Materials and methods

### 2.1 Building a database

We build a curated antibody database to train and test our method, comprising Fabs and VHHs. Fabs are composed of one constant and one variable domain for each of the heavy and the light chains. The two variable domains are denoted as the Fv variable fragment (Fv). VHHs are antibody fragments that correspond to the sole variable domain of the heavy chain. They represent a promising family for therapeutic antibodies.

The Fab data are originally fetched from *SabDab* (Dunbar *et al.* 2014) in the form of a list of protein chains. We fix a few broken annotations, notably for systems containing both Fabs and VHHs (more details in [Supplementary Section S1.2](#)). Using the PDB (Berman *et al.* 2000), we find the corresponding cryo-EM maps and download all corresponding maps and structures. Finally, we remove systems with

resolution below 10 Å or ones with no antibodies or antigen chains, yielding a total of 1032 maps. The maps include both Fabs and Fvs, but the constant region of Fabs is often missing in the deposited structure. To avoid false negatives in our data, we chose to always predict the position of the Fv as this is the only part consistently reported.

The resulting map files can be enormous (up to  $10^9$  grid cells) especially for symmetric assemblies, such as viruses, where the asymmetric unit only occupies a fraction of the map (e.g. pdb 7kcr). Since some regions of the map correspond to proteins omitted in the deposited structure, using whole maps would create artifacts of negative labels. To limit those artifacts in our dataset, we crop the original maps around the structure with a margin of 25 Å. Additionally, we resample maps to a fixed voxel size of 2 Å and normalize them by zeroing out negative values and dividing by the maximum of the map.

We split this dataset following a temporal splitting strategy with more recent systems in the test split (denoted as *sorted* setting). While this procedure is often used to provide a realistic use case, it can also introduce a bias, for instance toward structures of the spike protein of SARS-Cov-2 obtained during the COVID pandemic. Hence, we also report our performance following a random split (*random* setting). Given the stable performance across these splits and the high computational cost of training, we use a single random split. We obtain 722, 155, and 155 systems in the train, validation, and test splits, respectively. Note that a single system can include several Fabs. On average, there are 2.25 Fabs per system in our dataset. The number of Fabs per split is 1627, 320, and 374, respectively—with similar numbers for the random split. We now have a split database of cropped, resampled, normalized cryo-EM maps containing Fabs.

All of these steps are repeated to create a VHH database. We started from *SabDab-nano* (Schneider *et al.* 2022), applied the same filters on the raw data, resulting in 398 systems, and performed the random and sorted splitting

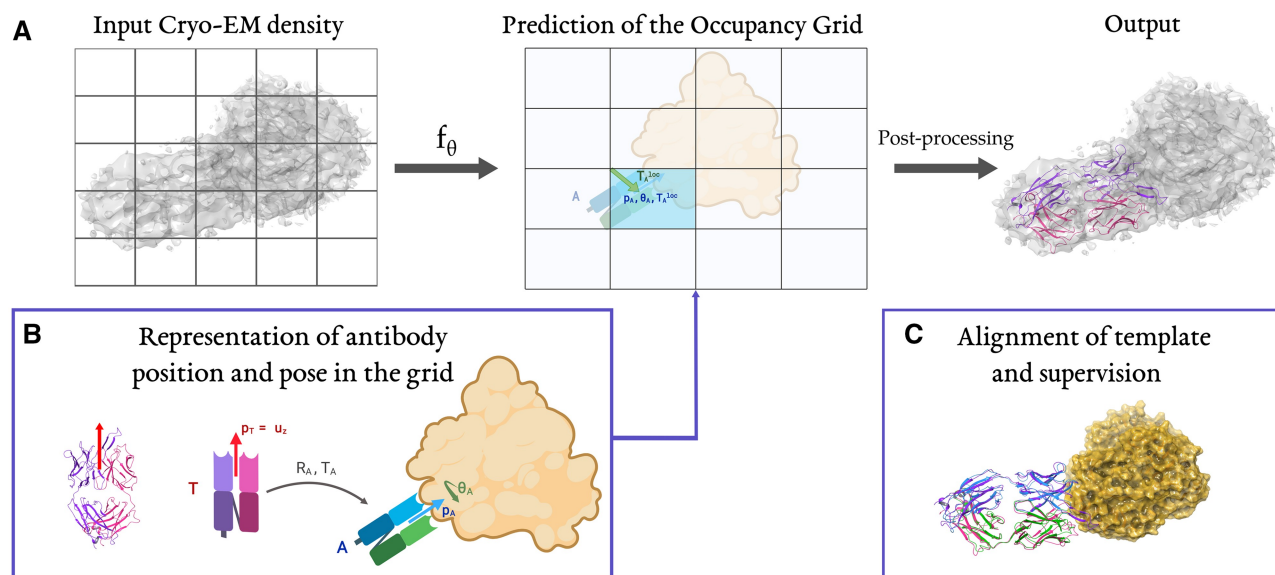
procedures as above. This amounts to 278, 60, 60 systems in each split containing 458, 74, 86 VHHs for a mean of 1.55 VHHs per system.

## 2.2 Overview and motivation

CrAI detects antibodies in cryo-EM maps using a customized deep learning-based technique, trained on our curated dataset comprised of 1430 cryo-EM maps containing Fabs and VHHs. When designing the model, we introduce a custom representation of the structure of antibodies to facilitate the learning process (Fig. 1B) and train a neural network to predict this representation. Once the model makes a prediction, persistence diagrams (Carlsson *et al.* 2004, Carlsson 2009) are used to select the relevant results which are transformed into a PDB file. Remarkably, at prediction time, this procedure does not require anything beyond the input Cryo-EM map. Our pipeline is illustrated in Fig. 1.

The design of our approach, CrAI, is motivated by several methodological challenges. First, due to the challenges inherent in the limited scale and significant noise present in the training data, we aim to incorporate prior information to make learning more data-efficient and robust. Specifically, we leverage the *conserved nature* of antibodies to approximate the detailed structure of the output by its *position* and *orientation*. Moreover, we use a custom parametrization of the rotation that takes into account a biologically *expected* pose, while being flexible to allow arbitrary orientations. Finally, the list of such representations of antibodies for a system is transformed into a grid overlaid over the cryo-EM map, such that the position of an antibody is encoded as an offset from a grid cell. The encoding of the output is shown in Fig. 1B.

Second, we introduce a fully convolutional design to accommodate arbitrary grid sizes that might be present at inference time. We also train the network with rotation augmentation to approximate rotation equivariance and accommodate arbitrary orientations.



**Figure 1.** (A) CrAI predicts an occupancy grid that represents antibodies found in an input map. The prediction of this occupancy grid can be post-processed into a PDB containing the predicted antibodies' structures. (B) The atomic structure of our template (pdb 71o8) is displayed next to its cartoon, with  $\vec{u}_z$  shown in red. We compute optimal alignments  $R_A^*$ ,  $T_A^*$  of our template onto Abs. We decompose the rotation into  $R_A^* = (\vec{p}_A, \theta_A)$  and the translations into a position in a grid  $T_A^{cell}$  and an offset from the grid corner  $T_A^{loc}$ . We thus obtain a grid with zero values except for cells containing an Ab, in those cells, we have  $\vec{p}_A, \theta_A$  and  $\vec{p}_A, \theta_A$ . (C) Example alignment of our template (red, purple) with the experimental structure of system 6bF9: antigen (orange) and Fab (blue, green). As can be seen, our template aligns well to other antibodies (RMSD = 1.8 Å).

Finally, we introduce a custom training loss that incorporates a formulation based on Optimal Transport along with persistence diagrams to better capture the geometric aspects of our problem, such as predicting nonoverlapping objects and including *distance*-based penalties between our prediction and the ground truth.

### 2.3 Problem formulation

Starting from an input cryo-EM map, we want our method to output the 3D coordinates of one or several Abs. Because of the highly conserved structure of Abs, we simplify our problem by only predicting how to align a fixed antibody template  $T$  (pdb 8fab) with the Abs, without deformations. We computed the optimal alignments with `pymol align` (DeLano 2002) during data pre-processing. We provide an example alignment of our template in Fig. 1A.

More formally: Let  $\mathcal{X}$  be the cryo-EM map we consider,  $n_{\mathcal{X}}$  the number of Abs contained in this map and  $\mathcal{A}_{\mathcal{X}} = \{A_i, 0 \leq i < n_{\mathcal{X}}\}$  the set of such Abs. Given an Ab  $A$  and a registration objective  $d$ , let  $R_A^*, T_A^* = \arg \min_{R, T \in (SO_3(\mathbb{R}) \times \mathbb{R}^3)} d(A, RT + T)$  be the translation and rotation that best align  $T$  to  $A$ . Finally let  $\mathcal{S}_{\mathcal{X}} = \{(R_{A_i}^*, T_{A_i}^*), A_i \in \mathcal{A}_{\mathcal{X}}\}$  be the set of optimal alignments. Note  $\mathcal{S}_{\mathcal{X}}$  consists of elements of the Euclidean group in 3D that is 6D, whereas elements of  $\mathcal{A}_{\mathcal{X}}$  are 3D coordinates for hundreds of atoms. In this article, we aim to predict  $\mathcal{S}_{\mathcal{X}}$  instead of  $\mathcal{A}_{\mathcal{X}}$ .

The optimal rotations mentioned above can be parameterized in many ways. Let  $\vec{p}_A$  denote the unit vector oriented from the center of mass of an Ab  $A$  toward its antigen. Since canonical binding tends to happen through the CDRs, we observe that it is easier to predict  $\vec{p}_A$  than the rotation around  $\vec{p}_A$ . Therefore, we decomposed  $R_A^*$  into a rotation transforming  $R_A^*$  into  $\vec{p}_A$  and one 2D rotation around  $\vec{p}_T$  of angle  $\theta_A$ . The generality of this decomposition is established in Supplementary Section S1.1 and its relevance is shown in Fig. 3B.

Our problem is now formulated as an object detection problem, which consists of detecting, localizing, and aligning the Abs in a given map. Following common practice in the object detection literature (Redmon et al. 2016), we overlay an occupancy grid  $\mathcal{G}^{\mathcal{X}, S}$  of size  $S$  over our input: this grid contains ones for cells encompassing an Ab and zeros elsewhere. The size of the cells of this grid corresponds to a fixed spatial volume and does not depend on the resolution of the map. We then decompose each translation into two parts: one going to the corner of the occupied cell and one from this corner to the Ab:  $T_A^* = T_A^{cell} + T_A^{loc}$ , as shown in Fig. 1A. Hence, finding the optimal translation comprises finding occupied cells and local translations in those cells. This decomposition makes the encoding of the output invariant to translation and avoids manipulating large values to encode translations in the grid.

### 2.4 Architecture and learning procedure

We aim to solve the object detection problem stated above with a Machine Learning approach, trained on our dataset. Given a cryo-EM map of size  $S$ ,  $\mathcal{X} \in \mathbb{R}^{S_i}$  and its corresponding occupancy grid of size  $S$ ,  $\mathcal{G}^{\mathcal{X}, S}$ , our network is a function  $f_{\theta} : \mathbb{R}^{S_i} \rightarrow \mathbb{R}^{10 \times S}$  such that  $\hat{y} = f_{\theta}(\mathcal{X}) \in \mathbb{R}^{10 \times S}$  is our prediction for  $\mathcal{X}$ . The prediction at a position  $s$  is denoted as  $\hat{y}^k(s) \in \mathbb{R}^{10}$ , where  $k = 1..10$ . The first dimension of this output,

$\hat{y}^0(\cdot)$ , is a prediction of the occupancy grid  $\mathcal{G}^{\mathcal{X}, S}$ . The nine other dimensions are predictions in each cell relative to the putative Ab contained in it. The details about both the exact role of each dimension as well as our training procedure are described below.

The architecture of our model  $f_{\theta}$  is a 3D UNet (Çiçek et al. 2016) with a depth of 4. To enhance robustness, the network is trained using data augmentation, with the eight possible rotations over a grid and random cropping inside the input grid up to three cells. We used Adam optimizer over 1000 epochs. Hyper-parameters and exact architecture were not extensively tuned to avoid artificially boosting performance. Finer details about the ones we used can be found in Supplementary Section S1.3.

### 2.5 Custom loss

#### 2.5.1 Prediction of the right cells using optimal transport

For convenience, in this section, we drop indices and denote the ground truth occupancy grid  $\mathcal{G}^{\mathcal{X}, S}$  as  $\mathcal{G}$ . The first slice of our output  $\hat{y}$  is a prediction of the occupancy grid and hence, let us denote it as  $\hat{\mathcal{G}}_s = \hat{y}^0(s)$ . In order to make  $\hat{\mathcal{G}}$  close to  $\mathcal{G}$ , we will use two loss terms.

Because most grid cells are unoccupied, our prediction is very imbalanced. Hence, our first loss term is a weighted binary focal loss (Lin et al. 2017) that focuses on cells with *wrong* predictions:

$$\text{focal}_{\gamma, \lambda}(\hat{y}, y) = \lambda y(1 - \hat{y})^{\gamma} \log(\hat{y}) + (1 - y)\hat{y}^{\gamma} \log(1 - \hat{y}),$$

$$\mathcal{L}_1(\hat{\mathcal{G}}, \mathcal{G}) = \sum_{s \leq S} \text{focal}_{\gamma, \lambda}(\hat{\mathcal{G}}_s, \mathcal{G}_s).$$

We observe that this focal loss does not consider the *distance* between our prediction and the ground truth: predicting the neighbor pixel results in the same loss value as predicting the opposite side of the map. To address this issue, we add an *optimal transport* term to our loss, denoted as  $\mathcal{L}_2$ . This term gives meaningful supervision to all voxels of our grids and depends on the distance to the closest occupied voxels.

After normalization, we can view  $\hat{\mathcal{G}}$  and  $\mathcal{G}$  as measures defined over the regular grid.  $\mathcal{L}_2$  is a regularized, corrected version of optimal transport called Sinkhorn divergence applied to our normalized predictions and targets. We computed this term with the *GeomLoss* (Feydy et al. 2019). We refer to Supplementary Appendix SA.4 and to Peyré and Cuturi (2019) and Feydy et al. (2019) for a more detailed discussion of the computation. The relevance of this loss is assessed in the Results section.

#### 2.5.2 Prediction within each cell

Let  $A$  be an antibody in our system and  $s_A$  its position in the grid. Beyond predicting the right grid cell, we also want finer grained prediction about its precise position in the cell  $T_A^{loc}$ , its orientation  $(\vec{p}_A, \theta_A)$  and its classification as a Fab or a VHH. Let  $\hat{y}_A^j = \hat{y}^j(s_A) \in \mathbb{R}^{10}$  be the prediction at this position, we will introduce additional loss terms to capture these finer grained predictions. We emphasize that these will only be applied on grid cells containing an antibody.

To predict the right pose of  $A$  in the cell, we use three values to predict the offset from the corner of the grid cell,  $T_A^{loc}$  learnt with a mean squared error loss  $\mathcal{L}_3$ . The following three values are used to predict  $\vec{p}_A$  by directly predicting its



coordinates. The corresponding loss,  $\mathcal{L}_4$ , is composed of a dot product term to control the direction of the prediction along with a term to make this vector unit norm. Favoring unit norms avoids numerically unstable normalizations. The remaining two values are used to predict the angle  $\theta_A$ . Instead of directly predicting the angle, we aim to predict  $u_A \in \mathbb{R}^2$ , the vector of polar coordinates  $(1, \theta_A)$ . This formulation avoids singularities and was shown to be beneficial when predicting angles (Jumper *et al.* 2021). Hence,  $\mathcal{L}_5$  has a similar form than  $\mathcal{L}_4$  in two dimensions to predict  $u_A$ . Using the notation  $\hat{y}_A^{j:j+k-1}$  to denote the  $k$  dimensional vector obtained from the concatenation of  $\hat{y}_A^j, \hat{y}_A^{j+1} \dots \hat{y}_A^{j+k-1}$ , we end up with the following losses:

$$\begin{aligned}\mathcal{L}_3(\hat{y}_A^{1:3}, T_A^{loc}) &= \text{mse}(\hat{y}_A^{1:3}, T_A^{loc}), \\ \mathcal{L}_4(\hat{y}_A^{4:6}, p_A) &= 1 - \langle \hat{y}_A^{4:6}, p_A \rangle + \text{mse}(\|\hat{y}_A^{4:6}\|, 1), \\ \mathcal{L}_5(\hat{y}_A^{7:8}, u_A) &= 1 - \langle \hat{y}_A^{7:8}, u_A \rangle + \text{mse}(\|\hat{y}_A^{7:8}\|, 1).\end{aligned}$$

Finally, as we use a single model for both Fabs and VHHs, we have a term that represents the probability that the object contained in the grid cell is a VHH and not a Fab. Let  $\delta_n(x)$  be the indicator function for VHHs (one if VHH else zero). We again construct a weighted focal loss, with a weight of  $\lambda_n = 1000/400$  corresponding to the ratio of VHHs to Fabs, yielding the last loss,

$$\mathcal{L}_6(\hat{y}_A^9, A) = \text{focal}_{\gamma, \lambda_n}(\hat{y}_A^9, \delta_n(A)).$$

To train our network we use a weighted sum of previous loss terms as the final loss. We sum the first two global ones and the sum of the four others over each antibody in our system:

$$\mathcal{L}_{tot}(\hat{y}, \mathcal{X}) = \sum_{i=1}^2 \mathcal{L}_i(\hat{y}, \mathcal{G}) + \lambda_s * \sum_{i=3}^6 \sum_{A \in \mathcal{A}_t} \mathcal{L}_i(\hat{y}_A, A).$$

We use values of 4, 30, and 0.2 for  $\gamma, \lambda$ , and  $\lambda_s$ , respectively, without extensive tuning, as they were empirically found to give good results.

## 2.6 Post processing

A well-known problem with object detection is the possibility that the network predicts overlapping objects. Given the size of occupancy grid cells (around 8 Å), adjacent cells could not both contain the center of mass of a Fab. Hence, high values for adjacent cells typically amount to the detection of the same underlying object. Non Maximal Suppression (NMS) algorithms are used to discard such redundant predictions. Starting with our grid  $\hat{y}_0$ , we want to obtain a list of the distinct local minima. In this paper, we used an approach based on *Persistence Diagrams* (PD), implemented with *crisper* (Chazal *et al.* 2013, Tralie *et al.* 2018, Kaji *et al.* 2020). Simply put, we decrease a threshold probability value from the maximum value of our grid  $\hat{y}_0$  and keep track of cells above this threshold. When the value of a cell goes over the threshold, either it has no neighbors in a visited connected component, giving *birth* to a new one, or all neighboring components are merged into the one with lowest initial values and others *die*. The difference between the values of death and birth are called lifetimes. We return connected components sorted by lifetimes.

This procedure takes into account both the *value* of a minimum and its *location* with respect to other minima. If we suppose that the number of objects to find is known, we keep proceeding until this number is reached (and refer to this setting as *num*). Otherwise we retain all predictions above a lifetime threshold. We use a threshold of 0.2 that was found to work best on the validation set (see [Supplementary Section S1.5](#)). We will denote this setting *thresh*. Interestingly, this procedure allows us to automatically detect the numbers of Abs in a map, without any prior information.

From our input map, we now have a list of predicted values. For each of those, we choose a template based on the classification in Fabs and VHHs and move this template to the predicted location and pose. We save the result as a file in PDB format.

## 3 Results

### 3.1 Baselines and metrics

Independent models were trained on the training sets obtained with the random and sorted splits (see Methods Section). We make inference of those models on their respective test sets, providing the network with the number of Abs to find (denoted as *num*) or relying on automatic thresholding to infer this number (*thresh*). For each system, a prediction method results in one or several predicted Ab positions. We report our results for individual Abs (*ab*) as well as aggregated by systems (*sys*) so that a system with many Fabs does not influence the results significantly.

When evaluating different approaches, individual predicted antibody positions need to be matched with actual antibody positions. This matching process is accomplished using the Hungarian algorithm (Kuhn 1955) on the distance between the center of mass of predicted and actual antibodies.

To the best of our knowledge, no tool enables predicting the position of antibodies solely from a Cryo-EM map. We benchmark against *dock\_in\_map* (Liebschner *et al.* 2019), a tool that takes the map along with the known atomic protein structures that need to be docked in the map. Considering that at test time, one does not have access to the ground truth structure, we ran *dock\_in\_map* with a fixed template Fab or Fv. Additionally, we ran *dock\_in\_map* with the actual structure giving us an upper bound of its performance.

The distribution of distances between centers of mass of the predictions and experimental structures shows that we outperform *dock\_in\_map*, even in the idealized scenario that uses the ground truth structure (see [Supplementary Fig. SA.2](#)). In the following, we will compare to this idealized and more challenging scenario as our baseline. Moreover, we observe that the predicted distances are bimodal: a first peak corresponds to systems predicted successfully and another spread mode corresponds to failed prediction.

We thus define a prediction as a true positive if it is closer than 10 Å to the ground truth. Performance is not very sensitive to the choice of this threshold as can be seen in the histogram. Undetected systems are false negatives and failed predictions are false positives. Our method's ability to detect antibodies is assessed using F1 scores and reported in [Table 1](#). This metric matches recall and precision for the baseline and the *num* settings since in those scenarios, any missed prediction (false negative) also results in an extra prediction (false positive). Recall and precision in the *thresh*

**Table 1.** Detection performance of the benchmark tool `dock_in_map` and of CrAI.<sup>a</sup>

|                          | random Split |              |             |             | sorted Split |             |             |             | Mean        |             |             |             |
|--------------------------|--------------|--------------|-------------|-------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                          | F1 (sys)     | F1 (ab)      | Distance    | RMSD        | F1 (sys)     | F1 (ab)     | Distance    | RMSD        | F1 (sys)    | F1 (ab)     | Distance    | RMSD        |
| Performance on Fabs      |              |              |             |             |              |             |             |             |             |             |             |             |
| <code>dock_in_map</code> | 66.6         | 61.9         | <b>0.78</b> | <b>1.87</b> | 71.3         | 66.0        | <b>0.68</b> | <b>1.69</b> | 69.0        | 64.0        | <b>0.73</b> | <b>1.78</b> |
| CrAI num                 | 97.3         | 96.7         | 1.40        | 6.21        | <b>96.9</b>  | <b>95.7</b> | 2.06        | 6.07        | <b>97.1</b> | 96.2        | 1.73        | 6.14        |
| CrAI thresh              | <b>98.1</b>  | <b>97.6</b>  | 1.39        | 6.36        | 96.1         | <b>95.7</b> | 2.03        | 5.83        | <b>97.1</b> | <b>96.7</b> | 1.71        | 6.10        |
| CrAI num + FitMap        | 93.1         | 91.2         | 1.12        | 5.41        | 93.8         | 92.0        | 0.83        | 4.12        | 93.5        | 91.6        | 0.98        | 4.77        |
| CrAI thresh + FitMap     | 93.8         | 91.9         | 1.15        | 5.44        | 93.6         | 92.4        | 1.15        | 3.79        | 93.7        | 92.2        | 1.15        | 4.62        |
| Performance on VHHs      |              |              |             |             |              |             |             |             |             |             |             |             |
| <code>dock_in_map</code> | 77.8         | 68.8         | <b>0.29</b> | <b>0.80</b> | 90.5         | 89.1        | <b>0.76</b> | <b>1.85</b> | 84.2        | 79.0        | <b>0.53</b> | <b>1.33</b> |
| CrAI num                 | <b>100.0</b> | <b>100.0</b> | 0.99        | 4.22        | <b>90.9</b>  | <b>90.6</b> | 2.32        | 7.59        | <b>95.5</b> | <b>95.3</b> | 1.66        | 5.91        |
| CrAI thresh              | 99.4         | 99.3         | 1.01        | 4.18        | 90.3         | 88.4        | 2.14        | 7.07        | 94.9        | 93.9        | 1.58        | 5.63        |
| CrAI num + FitMap        | 92.0         | 84.4         | 1.13        | 3.78        | 89.2         | 89.1        | 0.83        | 5.85        | 90.6        | 86.8        | 0.98        | 4.82        |
| CrAI thresh + FitMap     | 91.4         | 83.7         | 1.00        | 3.66        | 88.6         | 86.8        | 1.28        | 5.40        | 90.0        | 85.3        | 1.14        | 4.53        |

<sup>a</sup> We either provide the ground truth number of objects (num) or not (thresh), and optionally post process our results with FitMap. We report the F1 score on different data splits (random or sorted), aggregated by systems (sys), or not (ab). These include 155 systems containing 374 Fabs and 60 systems containing 86 VHHs. We additionally report the mean distance and RMSD of successful predictions. Best value in bold.

setting are reported in [Supplementary Table SA.1](#) and lead to a consistent analysis.

### 3.2 Antibody detection performance

CrAI *accurately finds Abs*. Regarding the prediction of Fabs, in the first two rows, note that CrAI drastically outperforms `dock_in_map` in terms of F1. This holds true in all settings for an overall F1 going from approximately 69%–97%.

The prediction of VHHs is more challenging because we have less training data (278 instead of 722 training examples), they display less canonical binding modes and are smaller and tend to be more buried into the map—as opposed to Fabs often sticking out. Despite those challenges, we maintain our performance on VHH, losing only three F1 score points, as can be seen in the top rows of the bottom part of the table.

CrAI *can be used with an unknown number of Fabs*. We now evaluate CrAI in the scenario where it also automatically estimates the *number* of Abs (num), and thus the network is given *only the map* without any additional information. This is to compare to the result of `dock_in_map` that is additionally given the *experimental structures of all antibodies* that are to be found in the map. As shown in the second and third lines of [Table 1](#), we retain most of the performance even in this more challenging, and more realistic setting. We find this remarkable, since the maps are highly heterogeneous and contain between one and six Abs and sets our method apart from *any* existing method in terms of its ability to detect antibodies in Cryo-EM maps in a fully automatic manner.

CrAI *runs fast, at all resolutions*. The average runtime of `dock_in_map` on our validation set is a prohibitive 883 s/system, which does not account for a few systems that we stopped after 5 h of computations. In comparison, our tool runs in 0.47 s/system, i.e. more than a thousand times faster when using a GPU (A40). Even when using only one CPU, our tool runs in 1.9 s/system, four hundred times faster than `dock_in_map` and fast enough for this computational step to integrate seamlessly in an analysis pipeline. This stems from the complexity of our algorithm, that is linear with regards to the grid size.

[Figure 2A](#) and [B](#) shows how our performance depends on resolution of the input map, over all of our datasets. Contrary to `dock_in_map`, we do not see a correlation and thus CrAI is *robust to low resolution*. This is a novel result as machine

learning methods for tracing such as ModelAngelo ([Jamali et al. 2024](#)) only work for resolutions better than 4 Å.

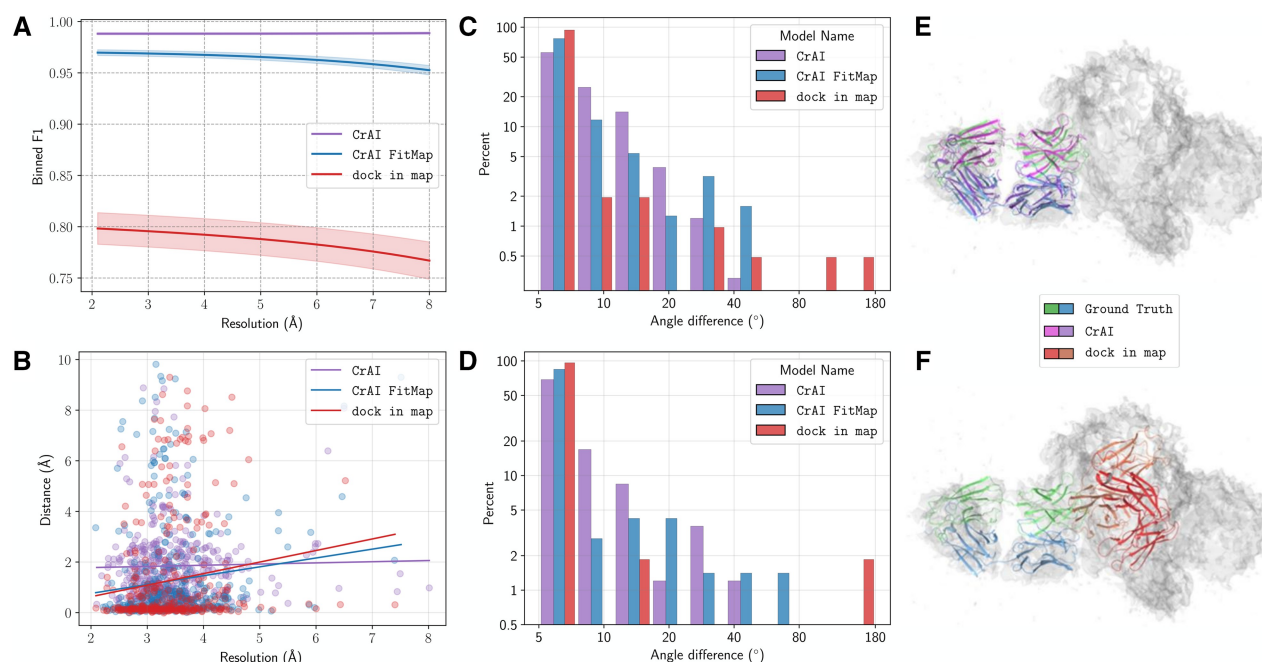
CrAI *predicts correct positions*. On successful predictions, the center of mass of the prediction is closer than 2 Å to the one of the ground truth, which is close to optimal considering the map resolutions. The observed RMSD are around 6 Å. Given a successful prediction, `dock_in_map` appears to be a bit more precise. However, this is expected as `dock_in_map` uses the ground truth structure instead of a template, which has a 1.6 Å RMSD to the ground truth on average. Moreover, distances are computed only over successful systems and thus include 20% *more systems in the CrAI column*.

To make our predicted positions more precise, one can first rapidly screen a map with a high F1 using CrAI, then precisely refine results in a local region using ChimeraX fast local refinement tool, FitMap. This additional step takes 0.4 s/system. It allows our prediction to have better distances and RMSD (4 Å), but slightly decreases performance and robustness to low resolution. This possibility is enabled in the proposed ChimeraX bundle.

CrAI *finds meaningful poses*. After validating the position of our predictions, we consider the predicted poses: are Abs in the correct orientation? Using the decomposition of rotations into predicting a vector  $\vec{p}_A$  and an angle  $\theta_A$ , we can compute the angles between predicted values and actual ones. We provide histograms for the distribution of these angles in [Fig. 2C and D](#), to show that most systems are predicted accurately. For the Fab data, the angle between  $\vec{p}_A$  and its prediction is on average of 7.8° and the one for  $\theta_A$  is 11.0°, low enough values to make the prediction almost overlap with ground truth. For the VHH data, those values are on average 7.5° and 9.7°, respectively. `dock_in_map` predicts more accurate orientations, which again stems from the use of the ground truth structure instead of a template. Post-processing with FitMap increases the pose accuracy on successful predictions but decreases it for low quality systems.

### 3.3 Further analysis

*Validation on true negatives*. We gather apo systems, using UniProt codes of antigen chains in our sorted test set, fetching other PDBs containing those codes and filtering them to not be included in SabDab or SabDab-nano, and to not be a virus, resulting in 26 apo systems.



**Figure 2.** In (A, B, C, and D), we compare the performance of our CrAI thresh model, optionally post-processed with FitMap and of dock\_in\_map. We report the binned F1 score (A) and the distance of successful prediction (B) as a function of the resolution of the systems. We also report the distribution of the angle between the predicted and experimental  $\vec{p}_A$  vector for Fabs (C) and VHHs (D). In the right column, we show the superimposition of the Fab from the PDB structure 7oh1 (blue/green) with the CrAI results (purple) (see E) and the dock\_in\_map results (red) (see F).

Although CrAI predicted that 12 systems contain an antibody, we found 8 of these systems actually included an antibody that was missing in SabDab, and have contacted the authors to include them in later releases (details in [Supplementary Section S2.2](#)). This results in a precision of 77% on empty maps. Some erroneous predictions occur when artifacts corresponding to small values occur in the map, and are not automatically discarded. Providing a reasonable threshold enhances the precision to 89%. We make this option available in the plugin.

**Case studies.** To provide a visual example of the prediction of our tool, we picked a low-resolution system containing a Fab (pdb 7oh1). This system represents the neurotoxin LC-HN domain in complex with TT110-Fab1 at a resolution of 8.0 Å ([Pirazzini et al. 2021](#)). After a failed crystallization attempt, authors obtained a cryo-EM structure that enabled them to identify the mechanism of action of a neutralizing antibody against viral infection with tetanus. The suggested Fab represents a promising lead for prophylactic and therapeutic use. We found out that CrAI correctly positions the Fab but dock\_in\_map misplaced it in the core of the antigen (see [Fig. 2E and F](#)).

We provide an extended analysis on five additional systems of the random split test set in [Supplementary Section S2.3](#), chosen because of their relevance in the context of drug discovery. Three systems contain Fabs and two include VHHs. The antibodies were found consistently and correctly by our tool.

**Ablation study.** To assess the relevance of different design choices of our approach, we retrain several models without specific individual features in our design. Since this means training a new model every time, we only perform this analysis in the Fab random split setting. We report the performance in the sys setting, but results are consistent in the ab setting. We try replacing Persistence Diagrams (PD) with a naive Non-Maximal Suppression (NMS) ([Felzenszwalb et al.](#)

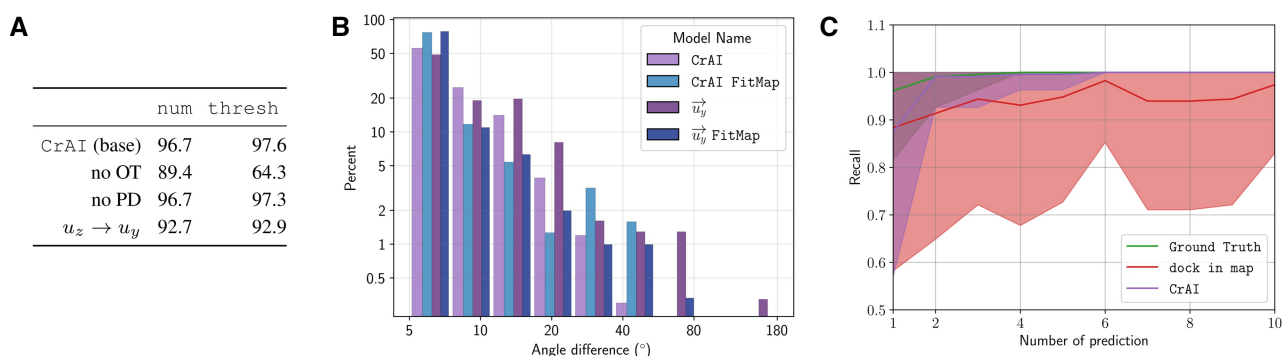
[2010](#), [Girshick et al. 2014](#)) that amounts to zeroing predictions around local minima. We also trained our model without using the Optimal Transport (OT) component, as is done in most classical object detection approaches ([Girshick 2015](#), [Redmon et al. 2016](#)). We also tried to keep the method fixed, but to disrupt the template encoding by decomposing the rotation using  $\vec{u}_y$ , instead of  $\vec{u}_z$ , as the main vector. We present the results in [Fig. 3A](#).

Persistence diagrams seem to enhance results in the thresh setting, with a limited impact. However, when removing optimal transport, performance collapses especially on the ability to predict the number of objects. This can be explained considering that antibodies cannot overlap (in contrast to detecting pedestrians in images for instance). Optimal transport helps to attribute a single detection to a region rather than enabling an arbitrary number of potentially overlapping/conflicting detections. Training the model with  $\vec{u}_y$  also significantly weakens detection performance.

Finally, we show the histogram of angular error of the model trained using  $\vec{u}_y$  in [Fig. 3B](#). It has an angle error of respectively,  $10.2^\circ$  and  $11.6^\circ$ , significantly higher on the vector prediction. Hence, predicting  $\vec{u}_z$  is easier than  $\vec{u}_y$ , which validates the nonrestrictive inductive bias that we introduced in our formulation.

**Failure analysis.** Since the number of failed systems with CrAI is relatively low, we visually inspected *all of them*. Upon inspection, we found that approximately one third of errors were actually expected, because the deposited PDB only reported one asymmetrical unit of the map (inducing artifacts of false positives), or had manually placed antibodies in bad map regions.

Moreover, we noticed that most of our actual errors originate from the ordering and thresholding of the predictions and not from the detection of antibodies. To further validate this observation, we performed a study of the recall of CrAI



**Figure 3.** (A) F1 score of our model and ablations on the random Fab split. We try removing the optimal transport loss, the persistence diagram post-processing or training to predict  $\vec{u}_y$  as a preferred axis instead of  $\vec{p}_A$ . (B) Distribution of angles of this model compared to our normal one. (C) F1 score of different approaches as a function of the number of predictions, in the VHH sorted setting. The solid line and shaded regions represent the mean and variance of the performance across systems.

and `dock_in_map` when forced to output a certain number of predictions. In Fig. 3C, we plot the fraction of VHHs captured by different approaches, when making a growing number of predictions per system. The green curve represents the best achievable results and does not always equal one as some systems contain multiple antibodies.

As can be seen in this figure, a small discrepancy exists between the ground truth and our tool. However, this discrepancy disappears around  $k=6$  predictions, suggesting that with this number of detections per system our method captures *all* VHHs. `dock_in_map` has a much wider gap that tends to stay consistent despite allowing it to output more predictions. This further justifies that most of our errors originate from the thresholding. Thus, if our tool fails to detect an antibody, practitioners can ask for more predictions with a high chance of seeing it predicted. A more in-depth failure analysis, along with visualization of the dominant error modes, recall curves for other splits and extensive plotting of all failed systems can be seen in [Supplementary Section S2.4](#).

## 4 Discussion

In this article, we addressed the problem of automatically finding the Abs in cryo-EM maps. This step currently constitutes a tedious, manual step thus hindering efficient and scalable structure estimation.

To achieve this goal, we proposed a customized solution, which exploits the structural properties and addresses the specific challenges of the problem, such as handling significant data scarcity and heterogeneity. Specifically, we leveraged the conserved structure of Fabs to cast this problem as an object detection one. We gathered and curated a database to enable a data-driven solution. Finally, we then designed a customized pose representation and loss based on optimal transport, which all help integrate prior information, while remaining efficient and flexible. Using our approach, no extra input is required to predict the number, position and pose of Fabs and VHHs in a map.

We validate our results on experimental maps and find the Ab positions with recall above 90%, which represents a 25% (resp 15%) improvement on Fabs (resp VHHs) over existing methods, while requiring no extra inputs and being *thousands of times faster* even when run on a single CPU. We show that the predicted pose correlates well with experiments, and illustrate our tool's performance on six systems relevant to drug design.

We believe that this tool can reduce the burden on structural biologists working with cryo-EM maps of Abs and accelerate the resolution of 3D complexes of Abs bound to their antigen. In line with this objective, the method ships as a ChimeraX (Pettersen *et al.* 2004) bundle to enable seamless integration.

In the future, it will be interesting to see if our approach can be pretrained on X-ray maps in a similar manner to (Karolczak *et al.* 2024), expanded to other conserved families, beyond Fabs and VHHs, and enable placing several folded domains into a map, which has so far been done without machine learning (Liebschner *et al.* 2019, Chang *et al.* 2022, Wang *et al.* 2024). More broadly, we could replace a fixed template by a parametrized family of embeddings of the folded domains, expanding the expressive power of our framework.

The exceptional throughput of our tool opens the door to finding Abs in the output of heterogeneous reconstruction or even continuous distribution of maps, and thus to capture several modes of antibody binding. Moreover, the automatization of cryo-EM structure resolution by our technique enables the generation of antibody-antigen complexes at a larger scale, a critical step to improve our understanding and detailed modeling of antibody binding. More structural information will open possibilities for better and more accurate antibody modeling that will speed up the drug discovery process of bio-therapeutics.

## Author contributions

Vincent Mallet (Conceptualization [equal], Data curation [equal], Formal analysis [equal], Investigation [equal], Methodology [equal], Resources [equal], Software [equal], Validation [equal], Visualization [equal], Writing—original draft [equal], Writing—review & editing [equal]), Chiara Rapisarda (Conceptualization [equal], Investigation [equal], Project administration [equal], Supervision [equal], Visualization [equal], Writing—original draft [equal], Writing—review & editing [equal]), Hervé Minoux (Conceptualization [equal], Formal analysis [equal], Project administration [equal], Supervision [equal], Validation [equal], Visualization [equal], Writing—original draft [equal], Writing—review & editing [equal]), and Maks Ovsjanikov (Conceptualization [equal], Formal analysis [equal], Funding acquisition [equal], Supervision [equal], Writing—original draft [equal], Writing—review & editing [equal])



## Supplementary data

Supplementary data are available at *Bioinformatics* online.

Conflict of interest: None declared.

## Funding

This work was supported by Sanofi. V.M. and M.O. were supported by DataIA, ANR AI Chair AIGRETTE and the ERC Starting Grant EXPROTEA. C.R. and H.M. were employees of Sanofi and may hold shares in the company. This work was granted access to the HPC resources of IDRIS under the allocation 2023-[AD010613356R1] made by GENCI.

## Data availability

The data underlying this article are publicly available in the PDB and SabDab databases. They can be processed, used to train and validate our model using our code, available in [github.com/Vincenxtx15/crIA-EM](https://github.com/Vincenxtx15/crIA-EM), and archived in Zenodo with DOI 10.5281/zenodo.14967869.

## References

- Arbabi Ghahroudi M, Desmyter A, Wyns L *et al.* Selection and identification of single domain antibody fragments from camel heavy-chain antibodies. *FEBS Lett* 1997;414:521–6.
- Bepko T, Morin A, Rapp M *et al.* TOPAZ: a positive-unlabeled convolutional neural network CryoEM particle picker that can pick any size and shape particle. *Microsc Microanal* 2019;25:986–7.
- Berman HM, Westbrook J, Feng Z *et al.* The protein data bank. *Nucleic Acids Res* 2000;28:235–42.
- Blundell TL, Patel S. High-throughput x-ray crystallography for drug discovery. *Curr Opin Pharmacol* 2004;4:490–6.
- Bouvette J, Huang Q, Riccio AA *et al.* Automated systematic evaluation of cryo-EM specimens with SmartScope. *Elife* 2022;11:e80047.
- Brzezinski D, Porebski PJ, Kowiel M *et al.* Recognizing and validating ligands with checkmyblob. *Nucleic Acids Res* 2021;49:W86–92.
- Carlsson G. Topology and data. *Bull Amer Math Soc* 2009;46:255–308.
- Carlsson G, Zomorodian A, Collins A *et al.* Persistence barcodes for shapes. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*. 2004, 124–35.
- Chang L, Wang F, Connolly K *et al.* Deeptracer-id: de novo protein identification from cryo-em maps. *Biophys J* 2022;121:2840–8.
- Chazal F, Guibas LJ, Oudot SY *et al.* Persistence-based clustering in Riemannian manifolds. *J ACM* 2013;60:1–38.
- Chiu ML, Goulet DR, Teplov A *et al.* Antibody structure and function: the basis for engineering therapeutics. *Antibodies* 2019;8:55.
- Çiçek Ö, Abdulkadir A, Lienkamp S *et al.* 3D u-net: learning dense volumetric segmentation from sparse annotation. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17–21, 2016, Proceedings, Part II* 19. Springer. 2016, 424–32.
- Cohen T, Halfon M, Schneidman-Duhovny D *et al.* Nanonet: rapid and accurate end-to-end nanobody modeling by deep learning. *Front Immunol* 2022;13:958584.
- Cushing VI, Koh AF, Feng J *et al.* High-resolution cryo-electron microscopy of the human cdk-activating kinase for structure-based drug design. *bioRxiv*, 2023, preprint: not peer reviewed.
- DeLano WL. The pymol molecular graphics system. 2002.
- Duggan S. Caplacizumab: first global approval. *Drugs* 2018;78:1639–42.
- Dunbar J, Krawczyk K, Leem J *et al.* Sabdab: the structural antibody database. *Nucleic Acids Res* 2014;42:D1140–6.
- Emmons C, Hunsicker L. Muromonab-cd3 (orthoclone okt3): the first monoclonal antibody approved for therapeutic use. *Iowa Med* 1987;77:78–82.
- Emsley P, Lohkamp B, Scott WG *et al.* Features and development of coot. *Acta Crystallogr D Biol Crystallogr* 2010;66:486–501.
- Fan Q, Li Y, Yao Y *et al.* CryoRL: reinforcement learning enables efficient cryo-EM data collection. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024.
- Felzenszwalb PF, Girshick RB, McAllester D *et al.* Object detection with discriminatively trained part-based models. *IEEE Trans Pattern Anal Mach Intell* 2010;32:1627–45.
- Feydy J, Séjourné T, Vialard FX *et al.* Interpolating between optimal transport and mmd using sinkhorn divergences. In: *The 22nd International Conference on Artificial Intelligence and Statistics*. 2019, 2681–90.
- Girshick R. Fast r-cnn. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, 1440–8.
- Girshick R, Donahue J, Darrell T *et al.* Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, 580–7.
- Hummer AM, Abanades B, Deane CM *et al.* Advances in computational structure-based antibody design. *Curr Opin Struct Biol* 2022;74:102379.
- Jamali K, Käll L, Zhang R *et al.* Automated model building and protein identification in cryo-em maps. *Nature* 2024;628:450–7.
- Jones PT, Dear PH, Foote J *et al.* Replacing the complementarity-determining regions in a human antibody with those from a mouse. *Nature* 1986;321:522–5.
- Jovčevska I, Muyldermans S. The therapeutic potential of nanobodies. *BioDrugs* 2020;34:11–26.
- Jumper J, Evans R, Pritzel A *et al.* Highly accurate protein structure prediction with alphafold. *Nature* 2021;596:583–9.
- Kaji S, Sudo T, Ahara K. Cubical ripser: software for computing persistent homology of image and volume data. *arXiv*, arXiv:2005.12692, 2020, preprint: not peer reviewed.
- Kaplan H, Crescioli S, Chenoweth A *et al.* Antibodies to watch in 2023. In: *MAbs*, Vol. 15, Taylor & Francis. 2023, 2153410.
- Karolczak J, Przybyłowska A, Szweczyk K *et al.* Ligand identification in cryoem and x-ray maps using deep learning. *Bioinformatics* 2024;41:btac749.
- Kimanius D, Dong L, Sharov G *et al.* New tools for automated cryo-EM single-particle analysis in RELION-4.0. *Biochem J* 2021;478:4169–85.
- Köhler G, Milstein C. Continuous cultures of fused cells secreting antibody of predefined specificity. *Nature* 1975;256:495–7.
- Kuhn HW. The Hungarian method for the assignment problem. *Naval Res Logist* 1955;2:83–97.
- Liebschner D, Afonine PV, Baker ML *et al.* Macromolecular structure determination using x-rays, neutrons and electrons: recent developments in Phenix. *Acta Crystallogr D Struct Biol* 2019;75:861–77.
- Lin T-Y, Goyal P, Girshick R *et al.* Focal loss for dense object detection. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, 2980–8.
- Matsumoto S, Ishida S, Araki M *et al.* Extraction of protein dynamics information from cryo-EM maps using deep learning. *Nat Mach Intell* 2021;3:153–60.
- McCafferty J, Griffiths AD, Winter G *et al.* Phage antibodies: filamentous phage displaying antibody variable domains. *Nature* 1990;348:552–4.
- Pearce NM, Krojer T, Bradley AR *et al.* A multi-crystal method for extracting obscured crystallographic states from conventionally uninterpretable electron density. *Nat Commun* 2017;8:15123.
- Peplow M. Cryo-electron microscopy makes waves in pharma labs. *Nat Rev Drug Discov* 2017;16:815–7.
- Pettersen EF, Goddard TD, Huang CC *et al.* UCSF chimera: a visualization system for exploratory research and analysis. *J Comput Chem* 2004;25:1605–12.

- Peyré G, Cuturi M. Computational optimal transport: with applications to data science. *FNT Mach Learn* 2019;**11**:355–607.
- Pfab J, Phan NM, Si D *et al.* Deeptimizer for fast de novo cryo-em protein structure modeling and special studies on cov-related complexes. *Proc Natl Acad Sci USA* 2021;**118**:e2017525118.
- Pirazzini M, Grinzato A, Corti D *et al.* Exceptionally potent human monoclonal antibodies are effective for prophylaxis and treatment of tetanus in mice. *J Clin Investig* 2021;**131**.
- Punjani A, Rubinstein JL, Fleet DJ *et al.* Cryosparc: algorithms for rapid unsupervised cryo-em structure determination. *Nat Methods* 2017;**14**:290–6.
- Qi CR, Litany O, He K *et al.* Deep hough voting for 3D object detection in point clouds. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, 9277–86.
- Redmon J, Divvala S, Girshick R *et al.* You only look once: unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, 779–88.
- Sanchez-Garcia R, Segura J, Maluenda D *et al.* MicrographCleaner: a python package for cryo-EM micrograph cleaning using deep learning. *J Struct Biol* 2020;**210**:107498.
- Scheres SH. Relion: implementation of a Bayesian approach to cryo-em structure determination. *J Struct Biol* 2012;**180**:519–30.
- Schneider C, Raybould MIJ, Deane CM *et al.* Sabdab in the age of biotherapeutics: updates including sabdab-nano, the nanobody structure tracker. *Nucleic Acids Res* 2022;**50**:D1368–72.
- Smith GP. Filamentous fusion phage: novel expression vectors that display cloned antigens on the virion surface. *Science* 1985;**228**:1315–7.
- TheAntibodySociety. The antibody society. therapeutic monoclonal antibodies approved or in review. 2023.
- Tralie C, Saul N, Bar-On R *et al.* Ripser.py: a lean persistent homology library for python. *JOSS* 2018;**3**:925.
- Wagner T, Merino F, Stabrin M *et al.* SPHIRE-crYOLO is a fast and accurate fully automated particle picker for cryo-EM. *Commun Biol* 2019;**2**:218.
- Wang F, Gong H, Liu G *et al.* DeepPicker: a deep learning approach for fully automated particle picking in cryo-EM. *J Struct Biol* 2016;**195**:325–36.
- Wang X, Zhu H, Terashi G *et al.* Diffmodeler: large macromolecular structure modeling for cryo-em maps using a diffusion model. *Nat Methods* 2024;**21**:2307–17.
- Yao R, Qian J, Huang Q *et al.* Deep-learning with synthetic data enables automated picking of cryo-EM particle images of biological macromolecules. *Bioinformatics* 2020;**36**:1252–9.
- Zhang J, Wang Z, Chen Y *et al.* PIXER: an automated particle-selection method based on segmentation using a deep neural network. *BMC Bioinformatics* 2019;**20**:41.
- Zhong ED, Bepler T, Berger B *et al.* CryoDRGN: reconstruction of heterogeneous cryo-EM structures using neural networks. *Nat Methods* 2021;**18**:176–85.
- Zou Z, Chen K, Shi Z *et al.* Object detection in 20 years: a survey. *Proc IEEE* 2023;**111**:257–76.