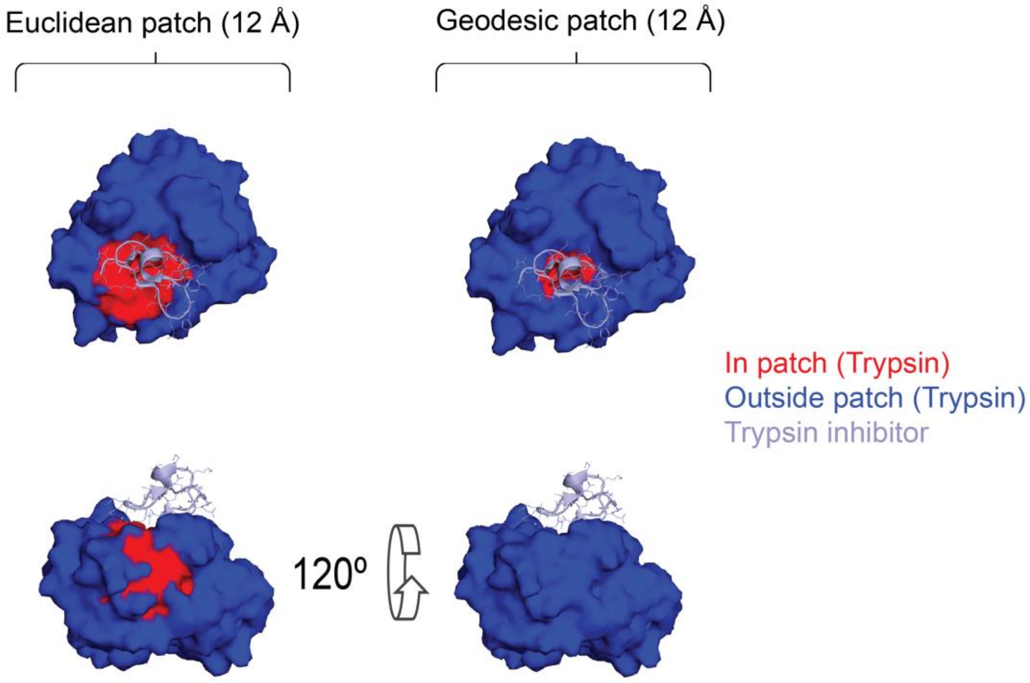
nature methods SUPPLEMENTARY INFORMATION
https://doi.org/10.1038/s41592-019-0666-6

In the format provided by the authors and unedited.

Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning

P. Gainza[1], F. Sverrisson[1], F. Monti[2,3], E. Rodolà[4], D. Boscaini[5], M. M. Bronstein[2,3,6] and B. E. Correia[1]*

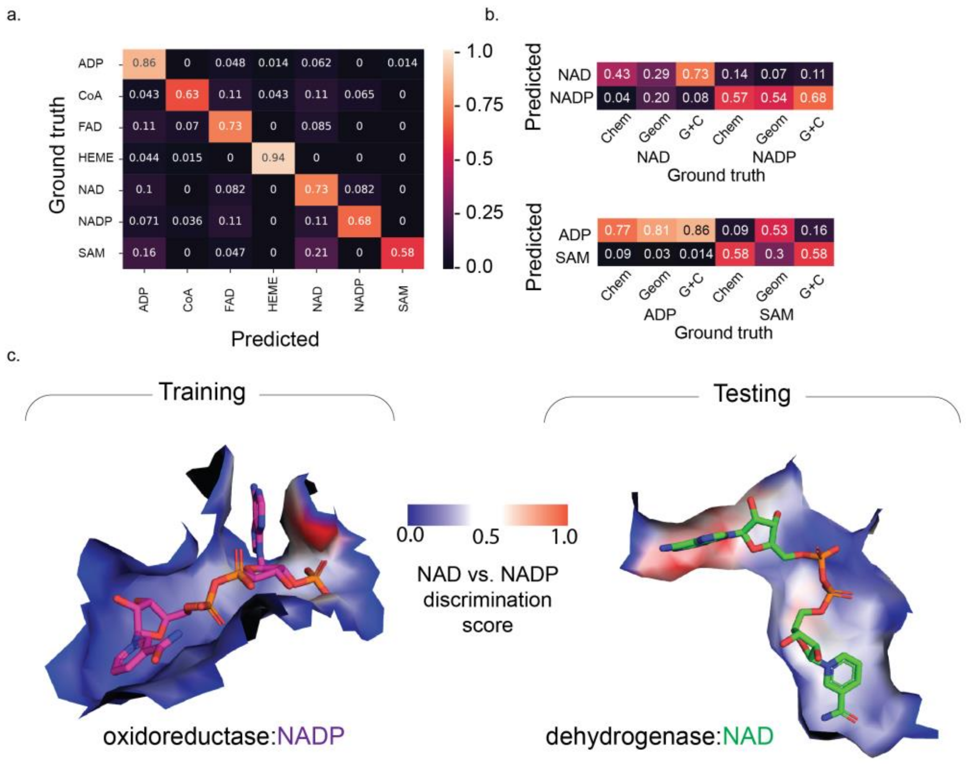
[1]Institute of Bioengineering, École Polytechnique Fédérale de Lausanne and Swiss Institute of Bioinformatics, Lausanne, Switzerland. [2]Institute of Computational Science, Faculty of Informatics, USI, Lugano, Switzerland. [3]Twitter, London, UK. [4]Department of Computer Science, Sapienza University of Rome, Rome, Italy. [5]Technologies of Vision Unit, Fondazione Bruno Kessler, Trento, Italy. [6]Department of Computing, Imperial College London, London, UK. *e-mail: bruno.correia@epfl.ch



Supplementary Figure 1

Example-based illustration on the importance of geodesic distances in modeling protein surfaces.

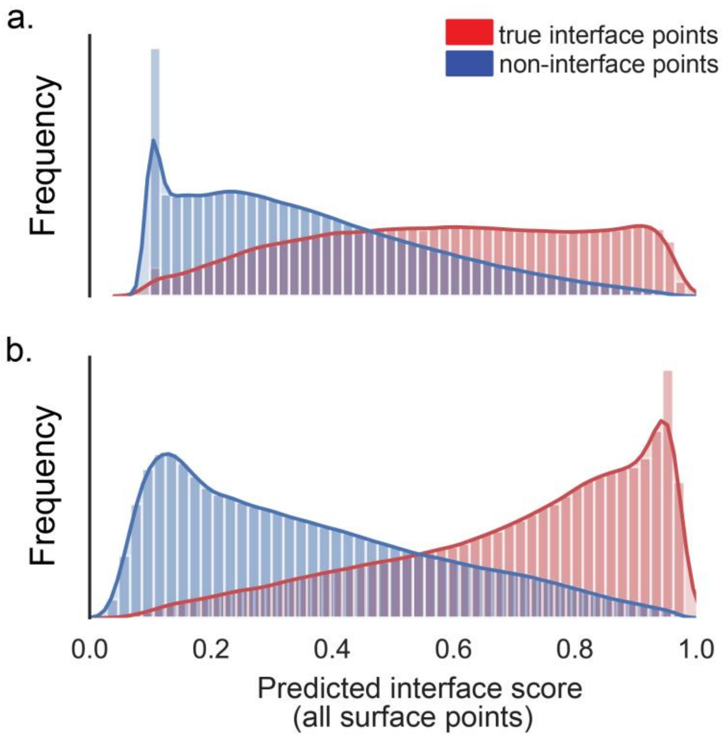
This example shows Trypsin (blue/red surface) in complex with the (cyan cartoon+line representation) (PDB ID 1PPE). We selected a point in the deep pocket of the interface, and colored in red every surface point within a 12 Å Euclidean radius-defined patch (left) or a 12 Å Geodesic radius-defined patch (right). The Euclidean patch (left, below) includes points on a different face of the protein, far from the binding site, while the geodesic patch only includes points in the face that interacts with the protein. This example shows that, especially in highly irregular surfaces the geodesic distances between points can be much larger than the Euclidean distances and that in such cases geodesic distances can be more relevant.



Supplementary Figure 2

Analysis of MaSIF-ligand performance for specific cofactors.

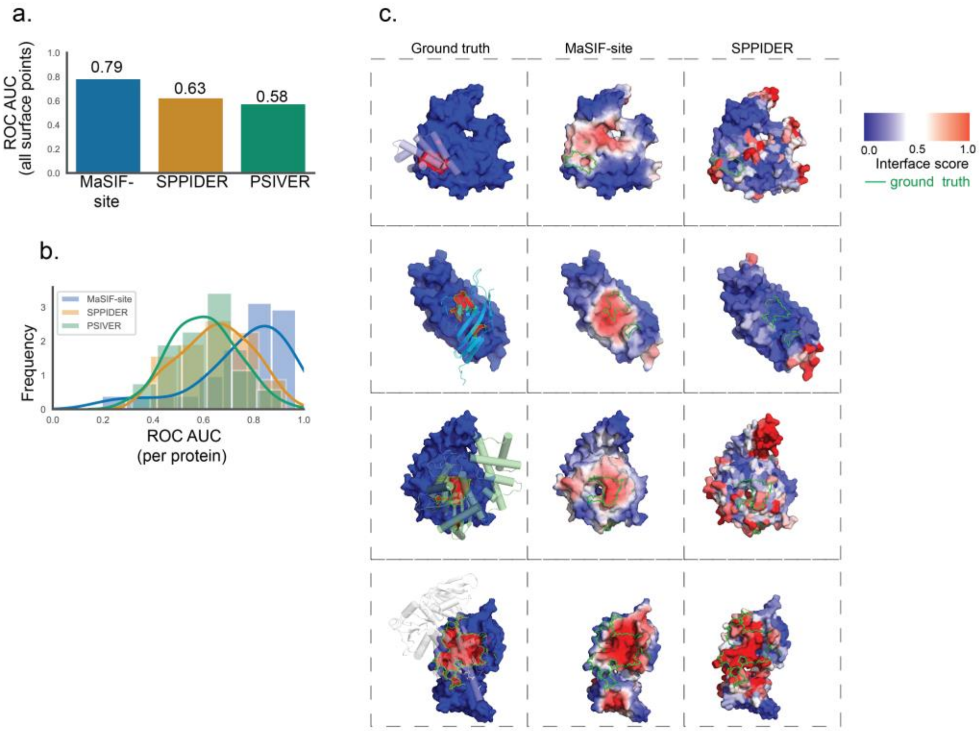
a. Confusion matrix of ligand specificity on a MaSIF-ligand neural network trained with all features. Number of pockets in each category: ADP:146, CoA:46, FAD:71, HEME:68, NAD:49, NADP:28, SAM:43. b. Subset of the confusion matrices showing the importance of the features in distinguishing pockets between highly similar ligands. Number of pockets in each category: ADP:146, NAD:49, NADP:28, SAM:43. c. Analysis of MaSIF-ligand's discrimination between NADP and NAD on two specific examples: a bacterial oxidoreductase and a human dehydrogenase. The bacterial dehydrogenase in the test set binds to NAD (PDB ID 2O4C), while its closest structural homologue in the training set corresponds to a mammalian oxidoreductase (PDB ID 2YJZ), which binds to NADP. Here we scored the pocket surface by a discrimination score, which scores each point in the protein surface by its weight in the neural network's distinction between NADP and NAD. Surface regions with high importance are shown in red, while those of low importance are shown in blue.



Supplementary Figure 3

MaSIF-site interface prediction score distribution for true positives (red) vs. true negatives (blue).

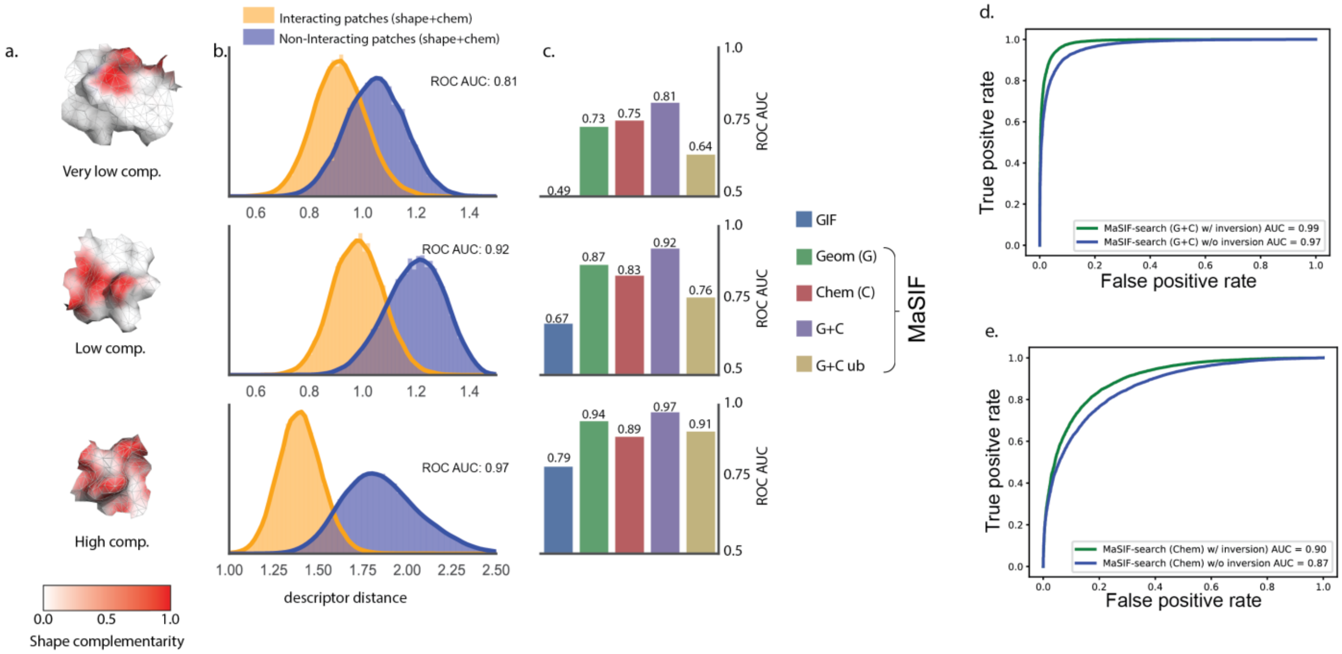
a. One convolutional layer obtains a ROC AUC value of 0.77 (n = 2192870 points from the test set) and b. Three convolutional layers obtain a ROC AUC value of 0.86 (n = 2192870 points from the test set).



Supplementary Figure 4

Comparison between MaSIF-site and two other predictors on a set of transient interactions.

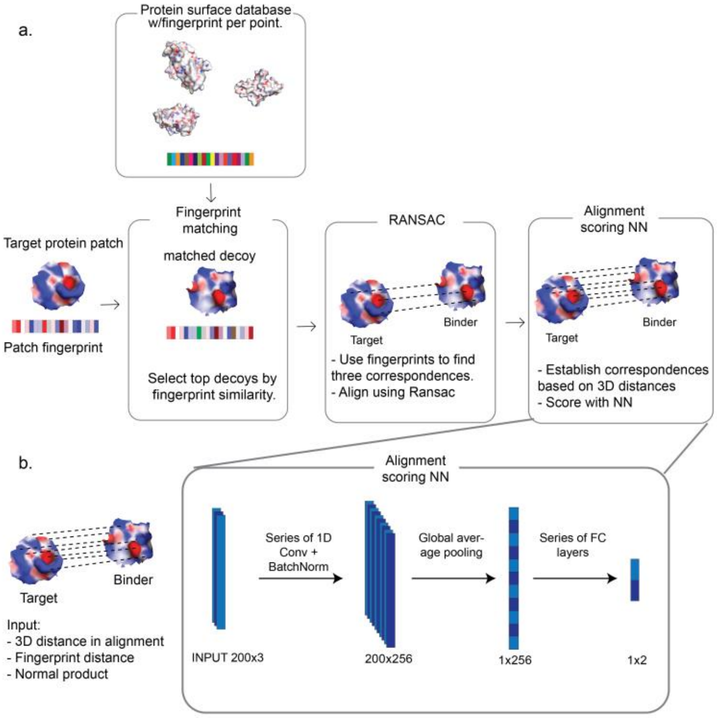
a. ROC AUC values over all surface points of MaSIF-site vs. SPPIDER vs. PSIVER on 53 proteins involved in transient interactions. b. Histogram showing the distribution of ROC AUCs per protein for the 53 proteins on a residue basis for MaSIF-site, SPPIDER and PSIVER. c. Randomly-selected examples from the testing set comparing MaSIF-site prediction with SPPIDER.



Supplementary Figure 5

Performance of MaSIF-search fingerprints under different shape complementarity filters for the interacting patches, and effect of inverting input features.

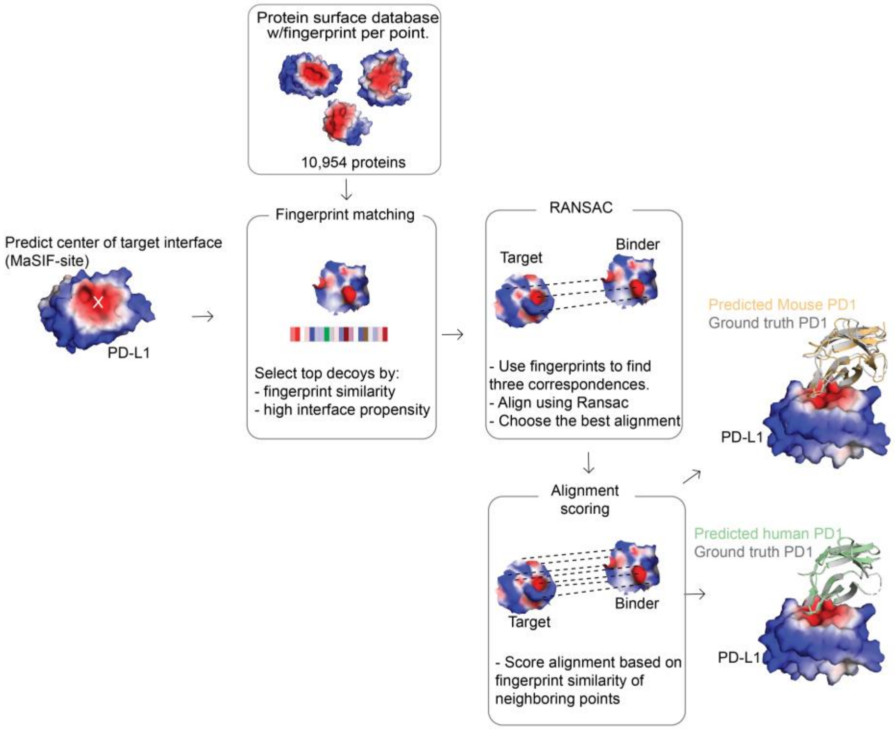
a. We set up three classes of interacting patches, filtered by shape complementarity, and trained neural networks with each set. The sets are illustrated here with three examples, where the surface is colored according to shape complementarity from white (0.0) to red (1.0). b. Descriptor distance distribution plot for interacting and non-interacting patches depending on the shape complementarity class. c. ROC AUC values for the GIF descriptors, MaSIF descriptors trained only on geometry, chemistry, or both, and patches found in unbound proteins within each complementarity class (G+C ub). # of pairs of patches: **high comp**, 38038 positives and 38038 negatives; low comp.: 16798 positives and 16798 negatives; low comp. 21297 positive and 21297 negatives. d-e. MaSIF-search benefits from the inversion of features in the input. d. ROC AUCs of a network trained/tested with inversion (green) vs. a network trained/tested without inversion (blue) using both Geometric (G) and chemical (C) features. The plot's ROC curve was computed on 13338 positive and 13338 negative pairs of samples. e. Performance of a network where electrostatics and the hbond features were inverted (green) vs. one in which they were not (blue), on a network trained with only chemical features.



Supplementary Figure 6

MaSIF-search protocol for the generation of protein complexes.

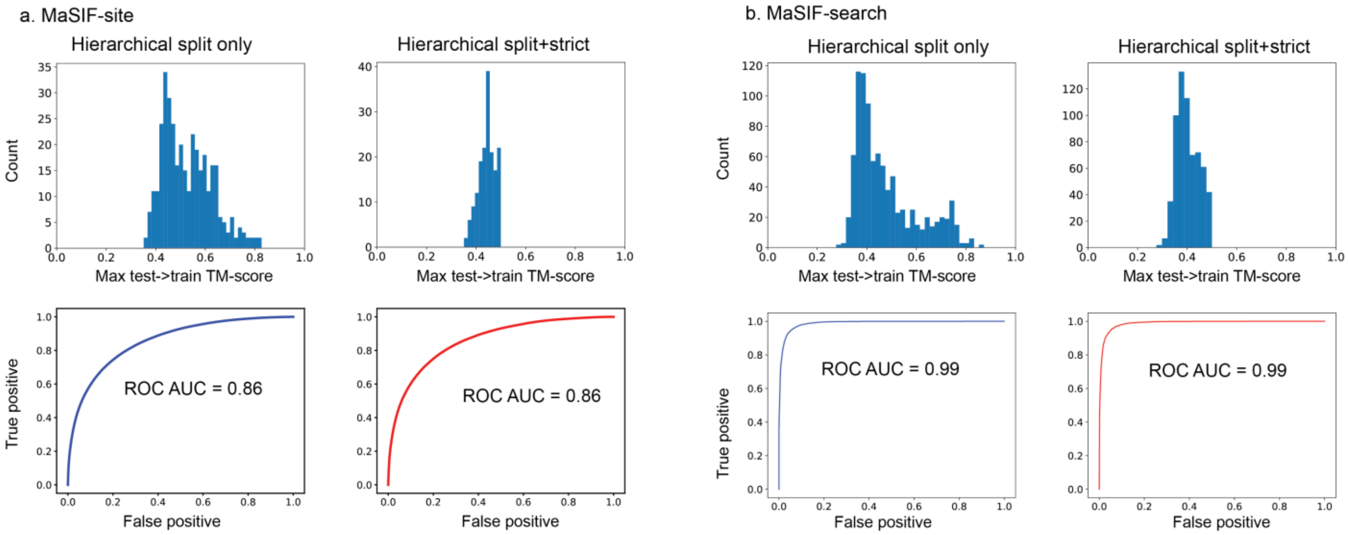
a. A fingerprint is computed on a selected target site (left). A database of proteins with precomputed fingerprints is searched for the K-most similar fingerprints. Once these are matched, a set of correspondences between the matched patches is found with the RANSAC algorithm, which uses the fingerprints of other points in the patch to obtain a good alignment. RANSAC selects the alignment with the most points within 1.5 Å of each other. The transformation is then scored using: Euclidean distances; fingerprint distances; and the normal products between neighboring points (see Methods). b. Neural network architecture for the alignment scoring function. Correspondences are first assigned between the aligned binder and target patches based on the nearest point in 3D space. For every correspondence, the 3D distance between the points, the Euclidean distance between the fingerprint descriptors and the product of their normals is input into the neural network. The input is a matrix of size 200 by 3: the maximum number of points allowed in the patch times the three features. The output is a 2-dimensional logit with the predicted score.


Protein surface database w/fingerprint per point.
10,954 proteins
Predict center of target interface (MaSIF-site)
X
PD-L1
Fingerprint matching
Select top decoys by:
- fingerprint similarity
- high interface propensity
RANSAC
Target
Binder
- Use fingerprints to find three correspondences.
- Align using Ransac
- Choose the best alignment
Alignment scoring
Target
Binder
- Score alignment based on fingerprint similarity of neighboring points
Predicted Mouse PD1
Ground truth PD1
PD-L1
Predicted human PD1
Ground truth PD1
PD-L1

Supplementary Figure 7

Hybrid MaSIF-search/MaSIF-site protocol to identify true binders against PD-L1.

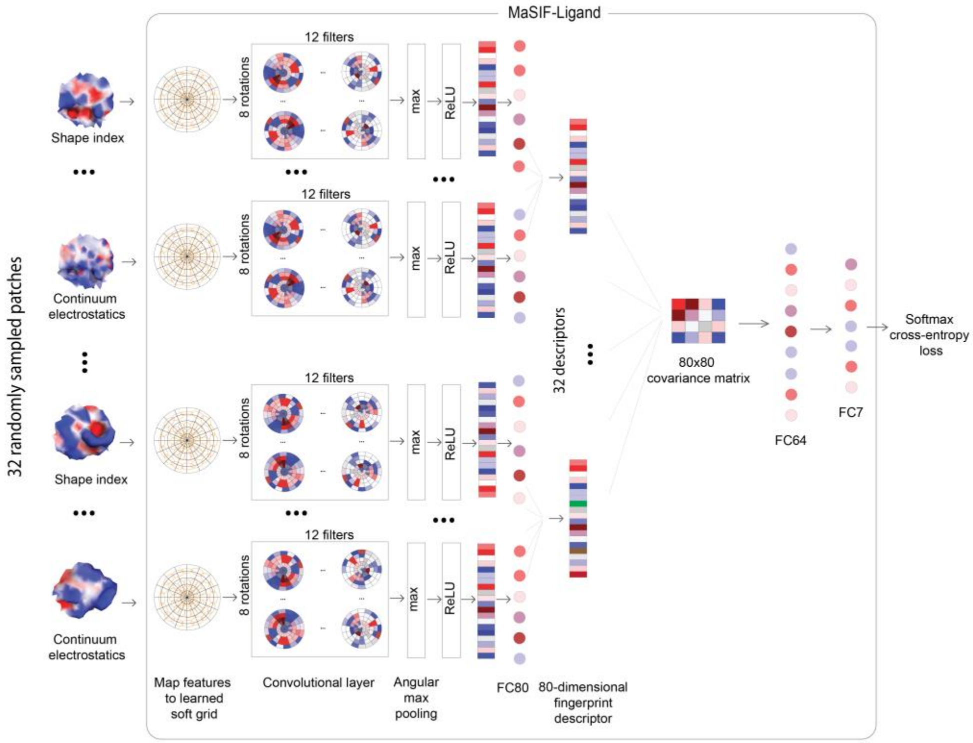
The target site is first predicted using MaSIF-site. Then a database of nearly 11,000 proteins is scanned, all patches with a MaSIF-site score > 0.9 and with a descriptor distance less than 1.7 are selected for alignments. Top candidates are matched using RANSAC, and reranked using the descriptor distance of all aligned points (described in Methods). The top predicted complex was the PD-L1:Mouse PD1 (PDB ID 3BIK), ranked #1 with an RMSD of 0.6 Å (shown here in pale orange). The PD-L1:Human PD1 (PDB ID 4ZQK), was ranked #8 with an RMSD of 0.3 Å. Both are shown overlaid over the initial complex (PDB ID 4ZQK). The entire runtime protocol took approximately 26 minutes (excluding descriptor precomputation time).



Supplementary Figure 8

The performance of MaSIF-search and MaSIF-site is not affected by a stricter structural split.

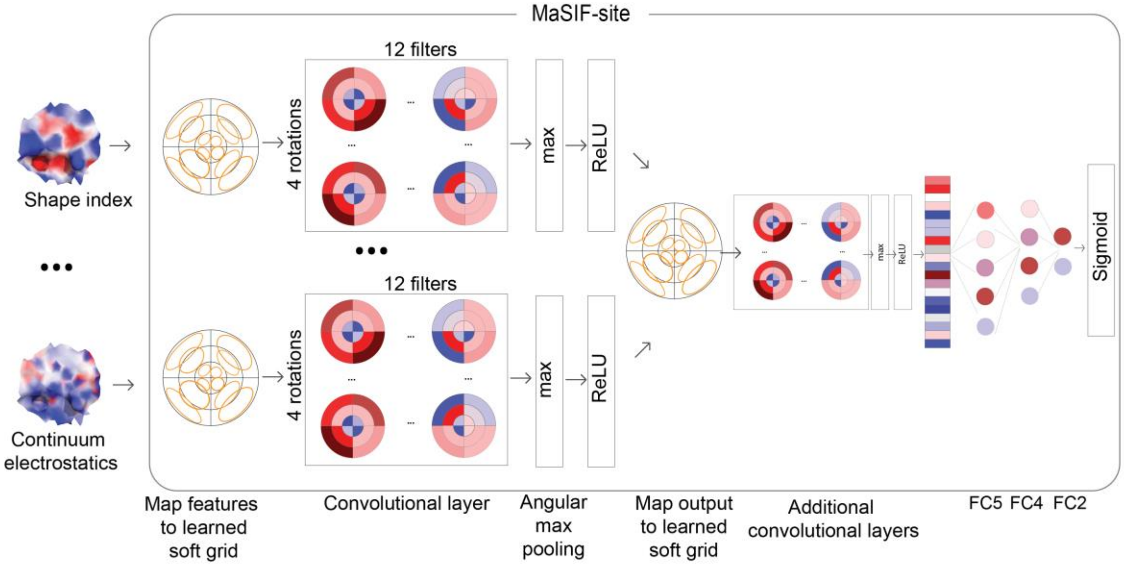
MaSIF-site and MaSIF-search's test sets were split from the training sets using a hierarchical clustering approach based on a matrix of TM-scores. In the case of MaSIF-search this split was performed using the interface TM-score. (hierarchical split only, a, b, top left). Some structures in the test set still maintain a TM-score above 0.5 to at least one member in the training set. (a,b, top right) We performed a stricter split by eliminating all members of the test set whose maximum TM-score to any member of the training set was above 0.5. (a,b, bottom right). The stricter split did not affect performance. a. MaSIF-site (left) Hierarchical split only test set consists of 359 proteins decomposed into 2191879 patches. (right) Hierarchical split+strict test set consists of 169 proteins decomposed into 1042951 patches. b. MaSIF-search (left) Hierarchical split only test set consists of a total of 957 proteins decomposed into 13338 interacting patch pairs and same number of non-interacting pairs. (right) Hierarchical split+strict consists of 635 proteins decomposed into 7135 interacting patch pairs and same number of non-interacting pairs.



Supplementary Figure 9

Network architecture for MaSIF-ligand.

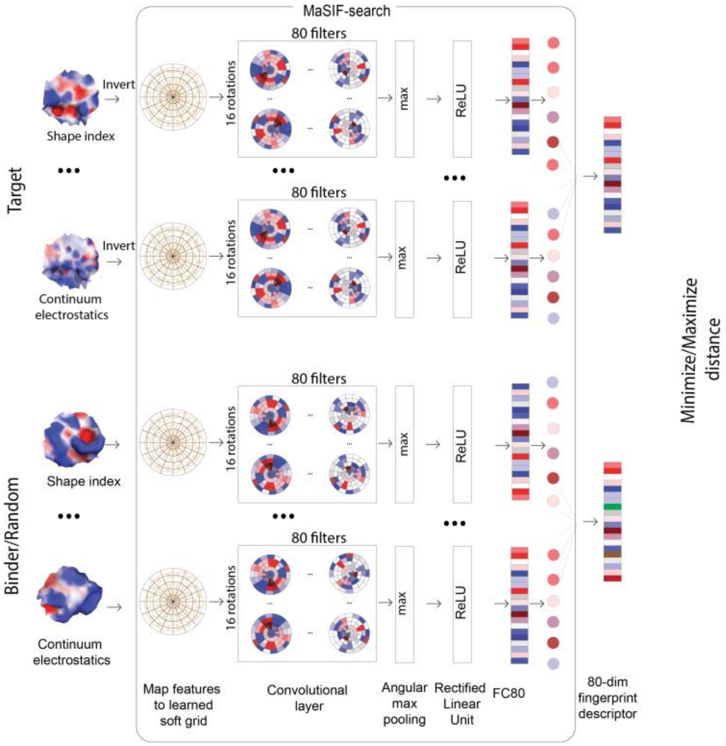
32 randomly sampled pocket patches are fed through convolutional layers followed by a fully connected layer (FC80). Descriptors are combined in a 80x80 covariance matrix followed by two fully connected layers (FC64 and FC7) and then softmax cross-entropy loss.



Supplementary Figure 10

Network architecture for MaSIF-site.

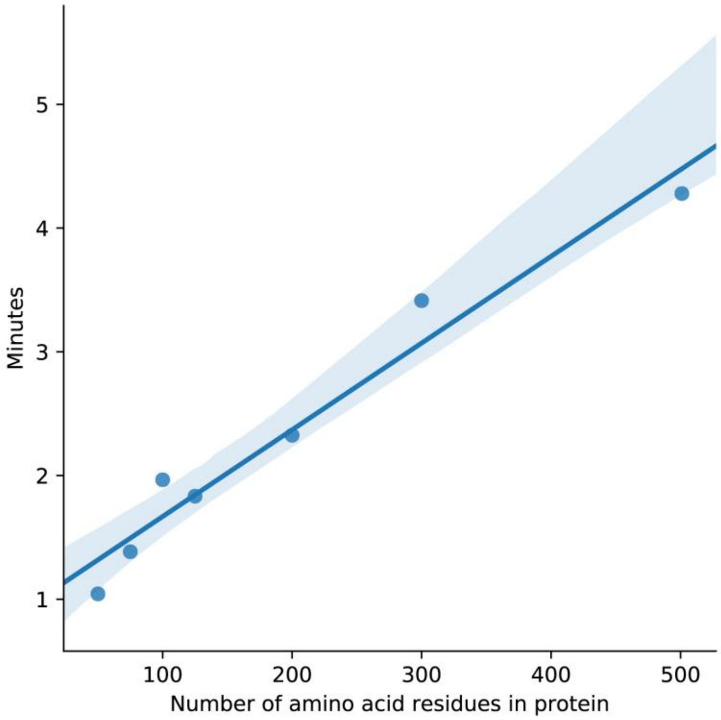
Patches are fed through convolutional layers followed by a series of fully connected layers (FC5, FC4, FC2), and finally a sigmoid cross-entropy loss.



Supplementary Figure 11

Network architecture for MaSIF-search.

Patches from the target and the corresponding binder or a random patch are fed through convolutional layers, followed by a fully connected layer (FC80). The L2-distance between the resulting descriptors is computed and the neural network is optimized to minimize this distance with respect to binder and maximize it with respect to the random patch.



Supplementary Figure 12

Total computation time for MaSIF-search and MaSIF-site for proteins of various sizes.

Proteins chains, of sizes: 50, 75, 100, 125, 200, 300, 500, were selected from the PDB. Each chain was run through both the MaSIF-site and MaSIF-search protocols, entailing: downloading the PDB, computing surfaces, input features, and coordinates, decomposing into patches, and computing MaSIF-site predictions and MaSIF-search descriptors. The y-axis shows the CPU user + System time + GPU time in minutes. GPU time consists of the time where the data is processed by the neural network, and was measured in real clock time (i.e. not GPU processor time). The total GPU time is low compared to the overall time, from 4 seconds for a 50-residue protein, to 12 seconds for a 500-residue protein. The line represents the regression fit to the n=7 data points and the shaded area represents the 95% confidence interval.

Supplementary note 1.

All structures in the Protein Data Bank (PDB; 16 Oct 2018) including a protein chain but no DNA or RNA were considered if they included any of these seven chemical identifiers: ADP, COA, FAD, HEM, NAD, NAP (for NADP) or SAM. This resulted in 1853 ADP structures, 490 COA, 2020 FAD, 4448 HEM, 1269 NAD, 1212 NADP and 393 SAM. After building the biological assembly of these structures, the dataset was filtered based on sequence identity, to reduce redundancy and similarity between structures in the training and test sets. The filtering was performed as follows: the PDB provides pre-computed sequence clusters based on 30% sequence identity; each protein structure in the dataset was associated with one or more of these clusters based on its protein chains; two protein structures were defined as homologous if the associated clusters of both proteins coincided. The dataset was then reduced by randomly sampling structures from the dataset, one at a time, while continuously eliminating their homologs from the sampling pool. This process resulted in a total of 1459 structures, which were then randomly assigned to training (72%), validation (8%) and testing (20%) sets. The surfaces for these structures were generated as described above, and patches of 12 Å radius extracted. If the center point of a patch was less than 3 Å from an atom for any of the seven ligands, the patch was labeled as a part of the binding pocket of the corresponding ligand.

Supplementary note 2.

From the NAD binding pocket of the dehydrogenase:NAD pocket (PDB id: 2O4C), 32 patch center points were randomly sampled 10000 times and binding predictions made for each, giving 10000 predictions (7-dimensional vectors). For each prediction the probability ratio NAD/NADP was computed. The predictions giving the top 90th percentile for this ratio were picked and the frequency of the patch centers behind these predictions were computed. The frequencies were normalized and overlaid on the protein surface. Same procedure was applied for the dehydrogenase:NADP complex (PDB id: 2YJZ) except that the NADP/NAD ratio was computed.

Supplementary note 3.

Comparison to KRIPO – KRIPO was used to generate fingerprints for ligand interactions in all structures from the training and testing set without fragmenting the ligands. Each fingerprint from the testing set was then compared to every fingerprint from the training set. KRIPO does not support the generation of fingerprints for HEME and thus this ligand was removed from the benchmark. Each fingerprint in the testing set was matched against ligand-labeled fingerprints in the training set (ADP, COA, FAD, NAD, NADP and SAM) resulting in six similarity scores for each query fingerprint. These scores were normalized to sum to one, giving a prediction of the ligand-binding preference.

Comparison to ProBiS – The ProBiS program was used to compute scores (*z-scores*) between each pocket in the test set to all pockets in the training set. For each test set pocket a score was assigned to each ligand type. The score for ligand X (X = ADP, COA, FAD, NAD, NADP and SAM) is the highest z-score found between the test set pocket and any pocket binding ligand X. We normalized the scores on a per-pocket basis as we found this improved ProBiS's ROC AUC value. The program was run with the -noprune flag to score all pockets, and the minimum z-score was set at -1000. To perform a pocket-level structural split (for the results shown in Fig 2e), all residues with an atom within 3.0 Å of a ligand atom were selected as the pocket residue. Then, TM-align was used to align each pocket of the test set to each pocket of the training set. Pockets aligning at TM-score > 0.5 to any element of the training set were eliminated from the structural split. The testing set consisted of all pockets that successfully ran on all three programs.

Supplementary note 4.

The PRISM database[56] of PPIs, a compendium of non-redundant PPIs found in crystal structures, was used as the first source. Proteins with parsing problems or that failed to complete the feature computation were discarded. The PRISM database contains many complexes formed by the contacting protein chains found in asymmetric crystal units, which likely do not form in solution. To remove those complexes, we discarded PPIs that have no pairs of patches below a minimum threshold of radial shape complementarity (set at S=0.4; see below for a definition). In total, 8466 proteins engaged in PPIs were taken from the PRISM database. In addition, 3536 non-obligate (transient) interactions were taken from three databases: the PDBBind[57], the SAbDab antibody:antigen database[58] and the ZDock benchmark set[59]. Finally, the resulting 12002 proteins were clustered according to sequence identity using the psi-cd-hit[60], at 30% sequence identity and one representative member was chosen from each cluster, resulting in 3362 proteins. A pairwise matrix of all TM scores for these proteins was then computed, and a hierarchical clustering procedure using scikit-learn (AgglomerativeClustering)[61] was used to split the sets, resulting in a training set of 3004 proteins and a testing set of 358 proteins. Using this hierarchical split approach still resulted in some members of the testing set having at least one member in the training set with a TM-score above 0.5. A TM-score above 0.5 means that the proteins assume roughly the same fold. However, upon performing a stricter split by eliminating all members of the testing set that align at TM-score > 0.5 to any member of the training set, we see no difference in performance (Supp. Fig 8).

Supplementary note 5.

Comparison to SPPIDER – The performance over a set of 53 single chains (from co-crystal structures) involved in known transient interactions for the test set was compared with that of the interface predictor SPPIDER[30]. Each protein was uploaded to the SPPIDER web site (http://sppider.cchmc.org/) and a regression-based prediction was computed on each residue. Following SPPIDER’s definition of ground truth interface residues[30] as closely as possible, the ground-truth interface residues were defined as those whose solvent excluded surface changes at least 5 Å^2 upon binding and at least 4% change in interface area. We note that we used the solvent-excluded surface for these calculations and not the solvent accessible surface. In order to perform a comparison with MaSIF-site, MaSIF-site's predictions were converted to a per-residue scoring by assigning the maximum score of all the residue’s points in the surface. A ROC AUC comparison on a surface point basis is shown in Supp. Fig. 4a.

Comparison to PSIVER - The sequence of each of the 53 proteins of the test set was uploaded to the PSIVER server (https://mizuguchilab.org/PSIVER/). The results of PSIVER assign a regression-based score on each amino acid residue of the protein, which was compared with the ground-truth. For both SPPIDER's and PSIVER's predictions in Figure 4, each of the designed proteins was assigned the predicted score as a b-factor in 1-99% scale and colored in PyMOL from a blue to red spectrum.

Supplementary note 6.

A dataset of protein pairs that were co-crystallized and shown to engage in PPIs were taken from the PRISM database (see above). In addition, 3536 non-obligate (transient) PPIs were taken as was done for the interface site prediction, forming a set of 6001 PPIs. For MaSIF-search we did not perform a sequence split, since we consider valid that two proteins with very high sequence identity (for example, two antibodies) binding to two completely different targets, can be in the training and testing set without the risk of overfitting. Instead, we perform our split using structural alignments of the interface atoms of each PPI. The PPI structural interface was extracted from the native complexes and a pairwise TM-align[63] score matrix with all interfaces was computed. Then, a hierarchical clustering of the structures was performed according to the TM-align score using scikit-learn's hierarchical clustering (AgglomerativeClustering)[61]. In total, the dataset was split into 4944 training PPI pairs and 957 testing PPIs. This list is complemented by 40 *apo* complexes, corresponding to those proteins in the testing PPIs such that both partners' *apo* crystal structure was available in the ZDock benchmark, belonging to the 'rigid docking' category[59]. The list of PDBs in the training and testing sets are provided in our github repository. Using this hierarchical split approach still resulted in some members of the testing set having at least one member in the testing set aligning with a TM-score above 0.5 to some member of the training set. However, upon performing a stricter split by eliminating all members of the testing set that align at TM-score>0.5 to any member for the training set, we see no difference in performance (Supp. Fig 8).

Supplementary note 7.

N=100 co-crystal structure complexes were randomly selected from the testing set. One of the two proteins was selected as the target protein; for each target protein, the patch with the highest radial shape complementarity to the binder protein patch in the co-crystal structure was selected as the target site (Fig. 5d). Each binder protein was docked onto each target site. The benchmark consisted of recovering the conformation of the true binder within a short list of the top-ranked results (top-100, top-10, top-1, shown in Fig. 5e). A second benchmark was performed with N=40 complexes in the *apo* state, aligned to the known bound complex. The benchmark for *apo* structures was performed in the same way as for the co-crystal structures, but the success criteria were relaxed to recover the conformation of the binder within a larger number of top results (top-1000, top-100, top-10). For all methods benchmarked, all binders were randomly rotated before performing any alignments.

Supplementary note 8.

Comparison to GIF descriptors - Geometric invariant fingerprint (GIF) descriptors were implemented to our best efforts according to the description by Yin *et al*[15]. For testing of the descriptors, the features of the target were inverted before computing the GIF descriptor.

Comparison to PatchDock - PatchDock[40] was used with default settings, assigning the residue closest to the target site as an active site residue. After all alignments, PatchDock's transformations for all targets were merged and ranked according to PatchDock's default Geometric Score. The top solutions (100, 10 and 1) for the bound complexes and unbound complexes (1000, 100, 10) were evaluated for agreement to the ground truth complex. PatchDock's time was measured as CPU usage time.

Comparison to ZDock - ZDock 3.0.2[41] was downloaded as compiled binaries for Linux 64-bits. The surfaces of each target and binder protein were first marked using the *marksur* program provided in the ZDock package. ZDock allows the definition of a target site, by allowing the user to 'block' every atom that is not in the target site. Thus, we determined the target site by drawing a 12 Å geodesic patch on the protein surface from the center of the interface. Then, all the atoms directly in contact with the vertices in the patch were added to a set of 'non-blocked' atoms. Every other atom in the protein was then blocked by setting the field in columns 55-56 of the target's pdb file to the code '19' as described in ZDock's user manual. For the bound (*holo*) benchmark, this process was run 10,000 times (100 binders for each of the 100 targets), while for the unbound (*apo*) benchmark the process was run 1600 times (40 binders for each of the 40 targets).

For each target:binder pair, ZDock generates, by default, 2000 docking results. Thus, the output files for all binders were merged and resorted by ZDock's score. Then, the top solutions for the bound complexes (100 ,10 and 1) and unbound complexes (1000, 100 and10) were evaluated for agreement to the ground truth complex.

Due to the large computational expense of these many runs, ZDock was run on a Google Cloud server with 96 virtual CPU processors and 360 GB of memory. The task was parallelized by running each target against all binders in its own thread. The time measured was CPU user time over all 10,000 runs for the *holo* benchmark, and over 1600 runs for the *apo* benchmark. Although the use of a different processor type could affect the running time comparisons with the PatchDock and MaSIF-search methods, the orders of magnitude difference between the methods is unlikely to vary significantly.

The output docking poses of ZDock[41] were used as input to ZRank2[43]. Although the running time of ZRank2 could be reduced by limiting the list of poses from ZDock, we used the entire list as the running time was still dominated by ZDock. The docked poses of the binders and targets were protonated with Reduce[51]. After ZRank2 was run, all the output results were merged and reranked according to the ZRank2 energy function. The time reported by ZDock+ZRank2 was the total CPU user time of ZDock + the total CPU user time for ZRank2. ZRank2 was also run on a Google Cloud server with 96 virtual CPU processors and 360 GB of memory. The task was parallelized by running each target against all binders in its own thread. The time measured was CPU user time over all 10000 runs for the *holo* benchmark, and over 1600 runs for the *apo* benchmark.

Supplementary note 9.

The task consisted on recovering the bound PD-L1:PD1 complex, among all possible complexes between PD-L1 and 10954 other proteins. First, the binding site scores on the surface of the PD-L1 chain (chain A in PDB id: 4ZQK[44]) was predicted using MaSIF-site. Then, the center of the interface was predicted by finding the patch with the highest mean interface score. Once the center of the interface was identified, the descriptor of this center point was matched to all patches in the 10954 proteins, for a total of 52 million fingerprints. Matches were ignored if the descriptor distance was greater than 1.7 or if the interface score was less than 0.9. The matches that passed this filter were explicitly aligned using our second stage alignment protocol. For this benchmark we used a simpler scoring function to rank each transformation, once correspondences were established between points (Supp. Fig 7), a score was computed according to the function $f = \sum_{ij} \frac{1}{d_{ij}^2}$ where d_{ij} is the descriptor distance between binder point i and target points j, such that i and j are within 1.0 Å. The top ten matches were then visually identified, showing the mouse PD1 (PDB id: 3BIK) as the top scoring match (ranked #1-#7), followed by the ground truth, wildtype match ranked #8.

Supplementary note 10.

The precomputing time of the PDB files to generate surfaces with features and runtime for MaSIF-search and MaSIF-site neural networks is dependent on the protein size, and is thus plotted in Supp. Fig. 12. For example, a 125 amino acid protein is processed in 99.4 s accounting CPU, System and GPU times. GPU times were measured using 'wall-clock' time, since standard UNIX time tools do not account for GPU processing time. All times were measured on an Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz, and an NVIDIA Tesla K40 GPU running Red Hat Enterprise Linux 7.4. PDB files precomputations were performed on CPUs, while neural network calculations were performed on GPUs.