A peer-reviewed version of this preprint was published in PeerJ on 3 February 2020.

View the peer-reviewed version (peerj.com/articles/cs-253), which is the preferred citable publication unless you specifically need to cite this preprint.

Sadique N, Ahmed AAN, Islam MT, Pervage MN, Shatabda S. 2020. Image-based effective feature generation for protein structural class and ligand binding prediction. PeerJ Computer Science 6:e253 https://doi.org/10.7717/peerj-cs.253

Image based effective feature generation for protein structural class and ligand binding prediction

Nafees Sadique [Equal first author, 1], **Al Amin Neaz Ahmed** [Equal first author, 1], **Md Tajul Islam** [1], **Md. Nawshad Pervage** [1], **Swakkhar Shatabda** [Corresp. 1]

[1] Department of Computer Science and Engineering, United International University, Dhaka, Dhaka, Bangladesh

Corresponding Author: Swakkhar Shatabda
Email address: swakkhar@cse.uiu.ac.bd

Proteins are the building blocks of all cells in both human and all our living creatures of the world. Most of the work in the living organism is performed by Proteins. Proteins are polymers of amino acid monomers which are biomolecules or macromolecules. The tertiary structure of protein represents the three-dimensional shape of a protein. The functions, classification and binding sites are governed by protein's tertiary structure. If two protein structures are alike then the two proteins can be of the same kind implying similar structural class and ligand binding properties. In this paper, we have used protein structure to generate effective features for applications in structural similarity to detect structural class and ligand binding. Firstly, we analyze the effectiveness of a group of image based features to predict the structural class of a protein. These features are derived from the image generated by the distance matrix of the tertiary structure of a given protein. They include local binary pattern histogram, Gabor filtered local binary pattern histogram, separate row multiplication matrix with uniform local binary pattern histogram, neighbour block subtraction matrix with uniform local binary pattern histogram and atom bond. The experiments were done on a standard benchmark dataset. We have demonstrated the effectiveness of these features over a large variety of supervised machine learning algorithms. Experiments suggest Random Forest is the best performing classifier on the selected dataset using the set of features. We believe the excellent performance of Hybrid LBP in terms of accuracy would motivate the researchers and practitioners to use it to identify protein structural class. To facilitate that, a classification model using Hybrid LBP is readily available for use at http://brl.uiu.ac.bd/PL/.

Protein-Ligand binding is accountable for managing the tasks of biological receptors that helps to cure diseases and many more. So, binding prediction between protein and ligand is important for understanding a protein's activity or to accelerate docking computations in virtual screening-based drug design. Protein-Ligand Binding Prediction requires three-dimensional tertiary structure of the target protein to be searched for ligand binding. In this paper, we've proposed a supervised learning algorithm for predicting Protein-Ligand Binding which is a Similarity-Based Clustering approach using the same set of features. Our algorithm works better than most popular and widely used machine learning algorithms

Image Based Effective Feature Generation for Protein Structural Class and Ligand Binding Prediction

Nafees Sadique[1,*], Al Amin Neaz Ahmed[1,*], Md Tajul Islam[1], Md. Nawshad Pervage[1], and Swakkhar Shatabda[1]

[1]**Department of Computer Science and Engineering, , United International University, Bangladesh***

Corresponding author:
Last Author[1]

Email address: swakkhar@cse.uiu.ac.bd

ABSTRACT

Proteins are the building blocks of all cells in both human and all our living creatures of the world. Most of the work in the living organism is performed by Proteins. Proteins are polymers of amino acid monomers which are biomolecules or macromolecules. The tertiary structure of protein represents the three-dimensional shape of a protein. The functions, classification and binding sites are governed by protein's tertiary structure. If two protein structures are alike then the two proteins can be of the same kind implying similar structural class and ligand binding properties. In this paper, we have used protein structure to generate effective features for applications in structural similarity to detect structural class and ligand binding. Firstly, we analyze the effectiveness of a group of image based features to predict the structural class of a protein. These features are derived from the image generated by the distance matrix of the tertiary structure of a given protein. They include local binary pattern histogram, Gabor filtered local binary pattern histogram, separate row multiplication matrix with uniform local binary pattern histogram, neighbour block subtraction matrix with uniform local binary pattern histogram and atom bond. The experiments were done on a standard benchmark dataset. We have demonstrated the effectiveness of these features over a large variety of supervised machine learning algorithms. Experiments suggest Random Forest is the best performing classifier on the selected dataset using the set of features. We believe the excellent performance of Hybrid LBP in terms of accuracy would motivate the researchers and practitioners to use it to identify protein structural class. To facilitate that, a classification model using Hybrid LBP is readily available for use at `http://brl.uiu.ac.bd/PL/`.
Protein-Ligand binding is accountable for managing the tasks of biological receptors that helps to cure diseases and many more. So, binding prediction between protein and ligand is important for understanding a protein's activity or to accelerate docking computations in virtual screening-based drug design. Protein-Ligand Binding Prediction requires three-dimensional tertiary structure of the target protein to be searched for ligand binding. In this paper, we've proposed a supervised learning algorithm for predicting Protein-Ligand Binding which is a Similarity-Based Clustering approach using the same set of features. Our algorithm works better than most popular and widely used machine learning algorithms.

INTRODUCTION

Protein tertiary structure comparison is very important in many applications of modern structural biology, drug design, drug discovery, in studies of protein-ligand binding, protein-protein interactions and other fields. This is especially significant because the structure of a protein is more protected than the protein sequence (Chothia and Lesk, 1986). Many works have been done to find protein binding (Brady and Stouten, 2000). Comparison of protein structure has been done in many works of literature by alignment of distance matrices (Holm and Sander, 1993), using iterated double dynamic programming (TAYLOR,

*First two authors contributed equally

1999), using elastic shape analysis (Srivastava et al., 2016) and many other techniques. The most common way of comparing protein tertiary structure is to treat the protein as a three-dimensional object and superimpose one on another. Different distances are used to calculate the differences between the proteins.

The distance matrix of α carbon can be seen extensively used in (Holm and Sander, 1997; Singh and Brutlag, 1997) as a feature which represents the tertiary structure of a protein chain. This feature is used as a feature vector which represents the structure of a protein to measure either similarity or dissimilarity to measure and compare the feature vectors with one another in pattern recognition literature. A mapped two-dimensional feature matrix is created from the 3D coordinate data of protein. The intra-molecular distance is used to make the α carbon distance matrix which mirrors the tertiary structure of a protein and the conserved elements of the secondary structure in it. With an input matrix size of N x N, the distance matrix based exact algorithms run in $O(N)$ time (Karim et al., 2015).

An image is basically a matrix of N x N dimension with corresponding data in each cell. Thus the distance matrix can be used as an image. Basically, three types of features can be generated from an image: pixel based, filter based and computationally generated features. Pixel-based features e.g histograms are simplistic and dependent on the capability of classification algorithms. Filter based methodologies transform the original image to use feature extraction methods. Refined algorithms are used to segment and other various algorithms are used to detect different features. Using ideas from computer vision and utilizing it in protein structure retrieval is not uncommon in the field. ProteinDBS server (Shyu et al., 2004) implement a similar approach in (Chi et al., 2005) by Chi et al. Texture features from the original size images and diagonally partitioned images were extracted by Chi et al. CoMOGrad and PHOG (Karim et al., 2015) also used images to extract their two novel feature whereas we are extracting histograms of local binary pattern images from the original image.

Human body uses protein for repairing tissues, making enzymes, hormones, and other biological chemicals. It is an essential building block of bones, muscles, cartilage, skin, and blood. On the other hand, a ligand is a material that has the potentiality to bind to and forms a composite with a biomolecule in order to carry out a biological function. In Protein-Ligand Binding, the ligand is usually a molecule which produces a signal by binding to a locus on a target protein. The binding typically results in a change of conformational isomerism (conformation) of the target protein. The evolution of the protein's responsibility depends on the development of specific sites which are designed to bind ligand molecules. Ligand binding ability is important for the management of biological functions. Ligand binding interactions changes the protein state and function. Protein-Ligand Binding prediction is very important in many applications of modern structural biology, drug design, drug discovery and other fields.

Many experimental techniques can be used to investigate various aspects of protein–ligand binding. X-ray crystallography, nuclear magnetic resonance(NMR), Laue X-ray diffraction, small-angle X-ray scattering, and cryo-electron microscopy provide atomic-resolution or near-atomic-resolution structures of the unbound proteins and the protein–ligand complexes, which can be used to study the changes in structure and/or dynamics between the free and bound forms as well as relevant binding events. Although experimental techniques can investigate thermodynamic profiles for a ligand–protein complex, the experimental procedures for determination of binding affinity are laborious, time-consuming, and expensive. Modern rational drug design usually involves the HTS of a large compound library comprising hundreds or thousands of compounds to find the lead molecules, but this is still not realistic using experimental methods alone. Different methods like Isothermal Titration Calorimetry (ITC) (Chaires, 2008), Surface Plasmon Resonance (SPR) (Patching, 2014), Fluorescence Polarization(FP) (Rossi and Taylor, 2011), Protein–Ligand Docking (Sousa et al., 2013), Free Energy Calculations (Steinbrecher and Labahn, 2010), etc are being used to predict ligand-binding prediction.

In this paper, we propose the combination of local binary pattern histogram, Gabor Filtered Local Binary Pattern Histogram, Separate Row Multiplication Matrix with Uniform Local Binary Pattern Histogram, Neighbour Block Subtraction Matrix with Uniform Local Binary Pattern Histogram and Atom Bond features to be used for protein similarity measurement. We extract the distance matrix of α carbon of a protein from PDB file and use the distance matrix as an image to extract our first four features and Atom Bond is extracted from the PDB files. We have used a large variety of classification algorithms to test the extracted features. We are also going to show the results and comparative study of different implementation methodologies such as wavelet and pyramid histogram based features (Ahmed et al., 2019) and CoMOGrad and PHOG. The method we have proposed is able to produce a better result on some classification algorithm over the previous methods on the same benchmark. In addition

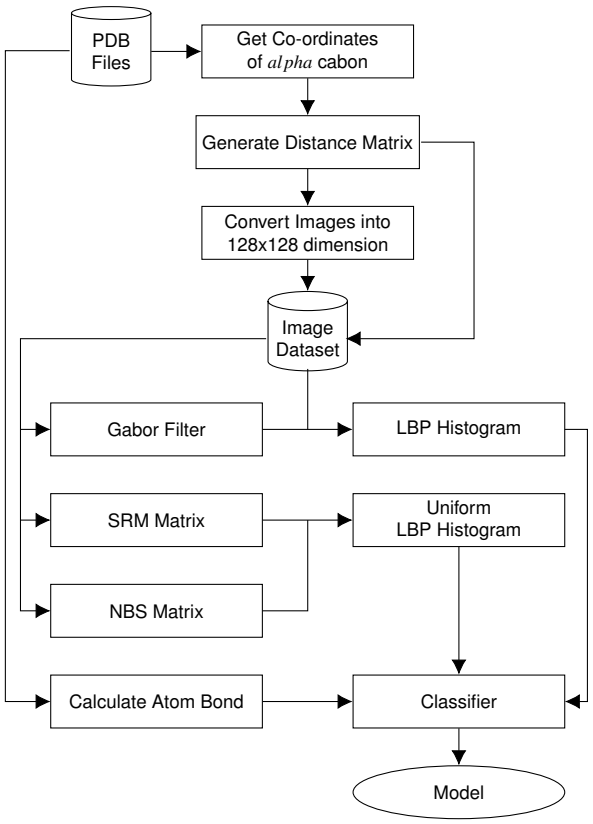

Figure 1. Block diagram of the methodology used in structural class prediction.

to that, we've proposed a supervised learning algorithm for predicting Protein-Ligand Binding which is a Similarity-Based Clustering approach using the same set of features. Our algorithm works better than most popular and widely used machine learning algorithms. Our proposed method uses the features proposed in this paper.

MATERIALS AND METHODS

Our methodology is divided into two parts. Firstly, we have generated image based features using protein tertiary structures and performed feature analysis based on the prediction power on the structural class prediction problem. In this section, we present the materials and methods for both of the problems. For each of the problems the dataset, features, necessary algorithms and performance measurement is described accordingly.

Structural Class Prediction

In this section, we present the methodology on structural class prediction. Atom bond features are generated from the protein tertiary structures given as PDB files. Images are created from the distance matrix calculated using α carbon atom coordinates of the amino acids of the protein structures in the given dataset. From each image of protein, we have derived five features. There are in total seven different classes of protein structures. Synthetic minority over-sampling technique (SMOTE) is used to handle class imbalance problem. K-fold cross-validation with three fold was used to test the capability and efficiency of the dataset. The block diagram of the methodology is given in Figure 1.

Structural Class Prediction Dataset

We have used 40 percent ID filtered subset of PDB-style files for SCOPe domains version 2.03 (Fox et al., 2013) as our dataset. It contains a total of 12119 PDB files. Each PDB files contains SCOP(e) concise classification string (sccs) which respectively describes class, fold, superfamily, and family. In this paper, we are going to experiment only with the class of the protein. In the dataset, there are total seven protein

Table 1. Protein Classes and its Corresponding Instances

Class Name	Total Instances
Small Proteins	640
All α Proteins	2195
α and β proteins(a/b)	3305
α and β proteins(a+b)	3006
Membrane and cell surface proteins and peptides	204
All β proteins	1485
Multi-domain proteins(α and β)	219

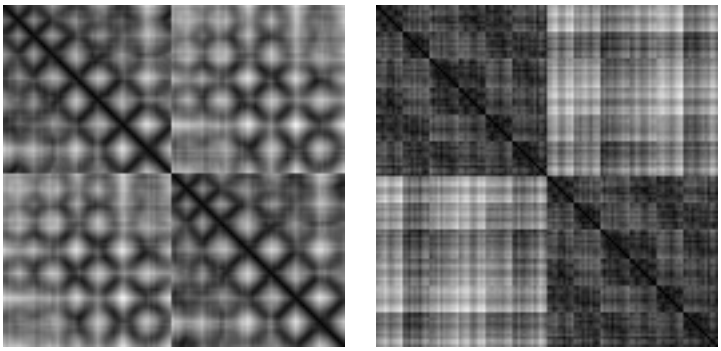

Figure 2. Sample images of protein structures after rescaling.

structural classes. For benchmark analysis with CoMOGrad and Phog, the common pdb files were used as dataset. The common PDB files are total of 11052. The details of the protein structural classes are given in Table 1. This dataset is widely used as a benchmark in the literature for protein structural similarity prediction (Karim et al., 2015).

Image Generation

We have generated images of protein structures according to the methodology described in CoMOGrad and PHOG (Karim et al., 2015). Only α carbons of the amino acids in the protein structure are considered for image generation. From the three dimensional coordinates of the α carbon atoms a distance matrix is generated by taking the Euclidean distance among all pairs. Thus only half of the image contains redundant information due to symmetry.

Scaling of Images

The dimension of protein images is based on the total number of α carbon they have. So, every individual protein images are different from the other in dimension. Therefore, the images were scaled to the same dimension. CoMOGrad and PHOG have used Bi-cubic interpolation and wavelet transform to scale all the protein images into 128 x 128 dimension (Karim et al., 2015). During the Bi-cubic interpolation step, most of the images were in 128x128 dimension so in the wavelet transform step they scaled all the images to that dimension. Thus, we have directly scaled the images to 128x128 dimension. We have used both real and scaled images to examine the differences in their predictive power. Sample rescaled images of protein structures are given in Figure 2.

Feature Extraction

We have generated five different feature groups. Our first four feature groups are different types of histograms and the fifth feature group is about the prognosis of the atoms. The histograms were taken from both scaled and unscaled images.

Local Binary Pattern Histogram Local binary pattern (LBP) histogram was first proposed by Ojala et al. (1994) and popularized by the work of Ojala et al. (2002). Local binary pattern computes the local representation of the texture of an image as a texture descriptor. Comparing each pixel with its

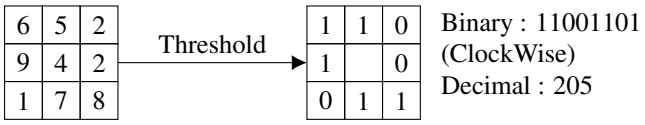

Figure 3. An example of basic LBP

neighboring pixels the local representation is created. The image is transformed into a grayscale image. In a 3×3 neighborhood, the center pixel value is calculated by comparing with its eight neighboring pixels. Each comparison gives a result of either 0 if the center pixel value is greater then the comparing neighbor pixel or 1 for the latter. A clockwise direction starting from the top-left one provides a binary number. The binary number is converted to a decimal number and the value is placed in the center pixel. LBP codes or Local Binary Patterns are the obtained binary numbers. An example of a basic Local Binary Pattern is given in Figure 3. After calculating the value for each pixel of the image, a histogram is calculated. A 3 x 3 neighborhood has $2^8 = 256$ possible patterns, thus the values range from 0 to maximum 255 in each pixel of the image. The total number of bins of the histogram is thus 256. We would get 256 attributes from each image. We have used zero padding technique to generate local binary pattern.

Gabor Filtered Local Binary Pattern Histogram (GfLBP-Hist) Gabor Filter is titled after Dennis Gabor. It is used for texture segmentation (Jain and Farrokhnia, 1991), optical character recognition (Jain and Bhattacharjee, 1992), edge detection (Mehrotra et al., 1992) etc. It is a linear filter which examines if there is any particular frequency content in the image in specific areas in a localized region throughout the point. The multiplication of a sinusoid and a Gaussian is called the Gabor filter (Eq.1).

$$g(x,y;\lambda,\theta,\phi,\gamma) = \exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda}+\phi\right) \quad (1)$$

.

Here, λ controls the wavelength of this sinusoid, θ is the angle of the normal to the sinusoid, ϕ is the phase shift of the sinusoid, γ controls the aspect ratio, The spatial envelope or the standard deviation of the Gaussian is σ. For our experiments, we have used $\lambda = 10$, $\theta = 0$, $\phi = 0$,$\gamma = 0.02$ and $\sigma = 5$. After applying the Gabor filter, LBP techniques are applied to the image to get 256 attributes.

Atomic Bond Features First of all, we've identified unique atoms amidst all the protein PDB files. From each protein PDB file, we've counted occurrences of each atom. Then we've taken the percentage as features of each atom among all the atoms that each protein has. Then we've taken first 100 sequential atoms and used their atomic mass as the feature. Then we've counted the bond that each pair of atoms has in a particular protein using atomic distance based on a threshold value. Finally, we've taken the percentage as the feature of the bond of each unique pair of atoms among all the bonds that the protein has.

Separate Row Multiplication Matrix with Uniform LBP Histogram(SRMMat-ULBP-Hist) The image is split into 3x3 matrices. From each matrix, we get 3 rows with the dimension of 1x3. By multiplying each row with the same 3x3 matrix, we get three result matrix consisting of 1x3 dimension. Each cell is divided by 100. The results are then put in the 3x3 matrix in accordance with the row numbers. The color intensity of an image is between 0 to 255. So, if the value of any cell of the result matrix is greater than 255, then the value is replaced with 255. After applying this technique, the uniform local binary pattern is applied. From Figure 4, (a) presents a 3x3 section of matrix and the rows, (b) exhibits the result of multiplication, (c) shows the value after dividing by 100, (d) shows the replacement result of value greater than 255 and (e) shows a 3x3 matix section after SRM-Matrix transformation.

Another variation of the LBP is called uniform pattern (Ojala et al., 2002). Some binary patterns occur more generally in texture images. If the binary pattern comprises of at most two 0-1 or 1-0 transitions when the bit pattern is held circular then the pattern is called uniform. For instance, 01000000 has 2 transitions, 00000111 has 2 transitions which are uniform pattern on the other hand 01010100 has 6 transitions,11001001 has 4 transitions which are not uniform. A neighborhood with the dimension of 3x3 has $2^8 = 256$ possible patterns with 58 of them being uniform. For estimating the histogram, every

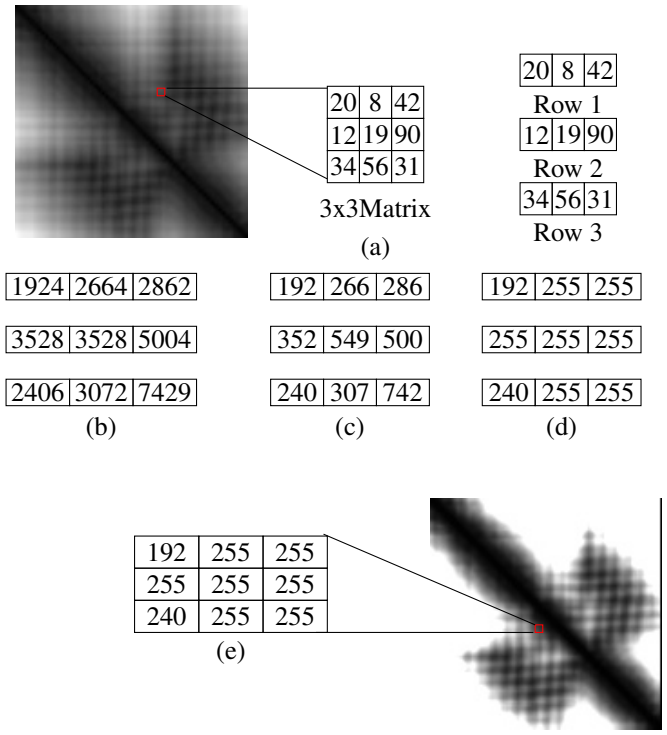

Figure 4. An example of Separate Row Multiplication Matrix with Uniform Local Binary Pattern Histogram.

Table 2. Feature Groups

Identifier	Feature Group Name	Number of Features
A	LBP-Hist	256
B	GfLBP-Hist	256
C	Atom Bond	116
D	SRMMat-ULBP-Hist	59
E	NBSMat-ULBP-Hist	59

uniform pattern gets a separate bin while a single bin is allotted for all non-uniform patterns. Therefore, from a uniform binary pattern, we get the histogram of total bin size of 59.

Neighbour Block Subtraction Matrix with Uniform LBP Histogram (NBSMat-ULBP-Hist) Blocks are of the same dimension, 3x3. Two blocks of matrices are considered neighbors for this method if the center cells are neighboring. Because of this, the value of the last two columns of the first block and first two columns of the second block are same. The two blocks of matrices are subtracted and the result is set in the place of the first block. If any of the cells have any negative number, then 0 is placed instead of the negative value. The replacing of value is made because the histogram bin begins from zero. Uniform local binary pattern is then used to compute the histogram.

Summary of all the feature groups used in this paper is given in Table 2.

Handling Imbalance in Data

From Table 1 it can be noted that the classes are imbalanced. To balance the classes, we have used Synthetic Minority Over-sampling Technique (SMOTE) (Chawla et al., 2002). The percentage of SMOTE indicates that how many more instances would be generated. As the highest number of instance a class has is 3305, we have over-sampled our instances close to that number. If x denotes the highest number of instances among all the classes and y denoted by a class which we will SMOTE then the expression for

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.27743v1 | CC BY 4.0 Open Access | rec: 19 May 2019, publ: 19 May 2019

the percentage calculation is $\frac{x-y}{y} * 100$. We have used 5 nearest neighbors to generate the over-sampled instances. After applying SMOTE to all data sets, the total number of instances in the dataset is 23132.

Classifiers Used

We have used five classifiers for the analysis of features applied to solve structural class prediction problem: K-Nearest Neighbor (KNN), Naive Bayesian Classifier, Support Vector Machines (SVM), Adaptive Boosting (AdaBoost) and Random Forest. A concise description of the classifiers is given in this section.

K-Nearest Neighbour (KNN) K-nearest neighbour algorithm (KNN) (Mohri et al., 2012) is a similarity-based classification technique. It is a lazy classification technique. Distance metrics are used for each instance of the whole dataset for calculating the K nearest neighbors. The labels of the nearest neighbors decide the label of the test instances. It works poorly for high dimensional data. Euclidean distance, Hamming distance, Manhattan distance, Minkowski distance, Tanimoto distance and Jaccard distance are used for similarity measures.

Naive Bayesian Classifier Naive Bayesian classifier (Mohri et al., 2012) is based on probabilistic inference of samples observed where the decision variable and the features form a very naive structure of Bayesian Network. Naive Bayesian classifiers work best for image recognition and text mining.

Support Vector Machine (SVM) Support Vector Machine (Mohri et al., 2012) works by creating and separating hyperplane for a given dataset by sampling different classes which are separated by maximum width.

Adaptive Boosting (AdaBoost) Adaptive Boosting classifier (Mohri et al., 2012) is a meta-classifier which aims to make a strong classifier using a set of weak classifiers. The classifiers whose performance are marginally better than random classifiers are called weak classifiers.

Random Forest Random Forest (Mohri et al., 2012) is an ensemble classifier.A decision tree is created in each iteration with features taken randomly. It samples selected features using bootstrap aggregating.

Ligand-binding Prediction

Protein Ligand Binding prediction is a binary class classification problem. We've used Image Based Features for each Protein and Ligand dataset. Our methodology learns threshold values from the training data and uses these in test data prediction. We have used the same set of features that were generated and analyzed for the structural class prediction problem to solve the ligang-binding problem. In this section, we present the necessary materials and methods that were used for the ligand binding problem.

Ligand-Binding Dataset

We've used Computer Vision and Pattern Discovery for Bioimages Group @ BII as our dataset. In our dataset, there are 3000 protein-ligand complexes that were determined experimentally with 3D structures available. Each protein and its ligand are of one-to-one correspondence, i.e. they can bind to each other and make Protein-Ligand complex. The dataset has 3000 pairs of protein and ligand where same name/ID of protein and ligand interacts/binds with each other.

We've used OpenCV (Bradski and Kaehler, 2008) library to create images from PDB files. For protein, we've considered the coordinates of only the alpha-carbons to generate the distance matrix to create image. Because alpha-carbon can represent the structural information of protein quite well. But the given ligands were small in terms of atom number. So, while creating ligand images, we've considered all the atom's co-ordinates for generating distance matrix.

Among the PDB files, 33 ligands have only one atom, which will create 1x1 image having no significance for feature extraction. So, we had to compromise those 33 ligands as well as 33 corresponding proteins from training.

Handling Imbalance

The given dataset has only positive instances (the pairs of protein and ligand where they bind with each other). But there were no negative instances (the pairs of protein and ligand where they do not bind with each other). The missing negative instances have created our dataset highly imbalanced. To overcome this imbalance, we've generated negative instances in two different ways.

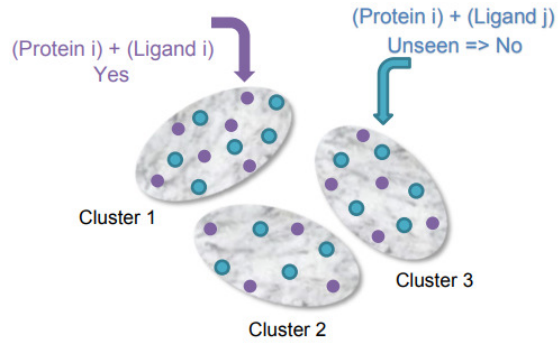

Figure 5. Clustering-Based Undersampling.

1. **Random Negative Undersampling:** We have 2967 protein PDB and 2967 ligand PDB where 8803089 pairs are possible. Among these, 2967 pairs are given as positive instances and the rest 8800122 pairs are unknown/unseen instances. From the unseen pairs, we've taken 2967 pairs randomly as negative instances to make our dataset balanced.

2. **Clustering-Based Undersampling:** Using the positive instances (2967 pairs), we've created 10 clusters. Then we've searched for 2967 unseen pairs randomly as negative instances where they belong to those 10 clusters. We've made sure that each cluster has exactly same number of positive and negative instances to make the dataset balanced (See Figure 5).

Similarity Based Classifier

We've developed a similarity-based clustering method to predict the binding class. Distance is used to measure similarity. Our methodology is given in Figure 6 and the pseudo-code in Algorithm 1.

Data: A pair (p, l), a protein structure and ligand tructure in pdb format
Result: Decision, whether they will interact or not
for *all proteins and ligands* **do**
  generate images & extract features
end
for *each of the given pairs of protein-ligand* **do**
  $\mathbb{NP} \leftarrow k$-NEARESTPROTEINS(p) of the given protein
  $\mathbb{RL} \leftarrow k$-RELATEDLIGANDS$(\mathbb{NP})$
  $d_l \leftarrow$ distance between given ligand, l & $\mathbb{RL}$
  if $d_l < threshold_l$ **then**
    $v_l \leftarrow$ vote for positive bind
  else
    $v_l \leftarrow$ vote for negative bind
  end
  $\mathbb{NL} \leftarrow k$-NEARESTLIGANDS(l) of the given ligand
  $\mathbb{RP} \leftarrow k$-RELATEDPROTEINS$(\mathbb{NL})$
  $d_p \leftarrow$ distance between given protein, p & $\mathbb{RP}$
  if $d_p < threshold_p$ **then**
    $v_p \leftarrow$ vote for positive bind
  else
    $v_p \leftarrow$ vote for negative bind
  end
  $v \leftarrow$ weighted majority voting between (v_l, v_p)
end
return v

Algorithm 1: Similarity based clustering algorithm.

From the PDB dataset of proteins and ligands, firstly we have generated images and converted to 128×128 images for each protein and ligand. From these images we have generated 2 different features.

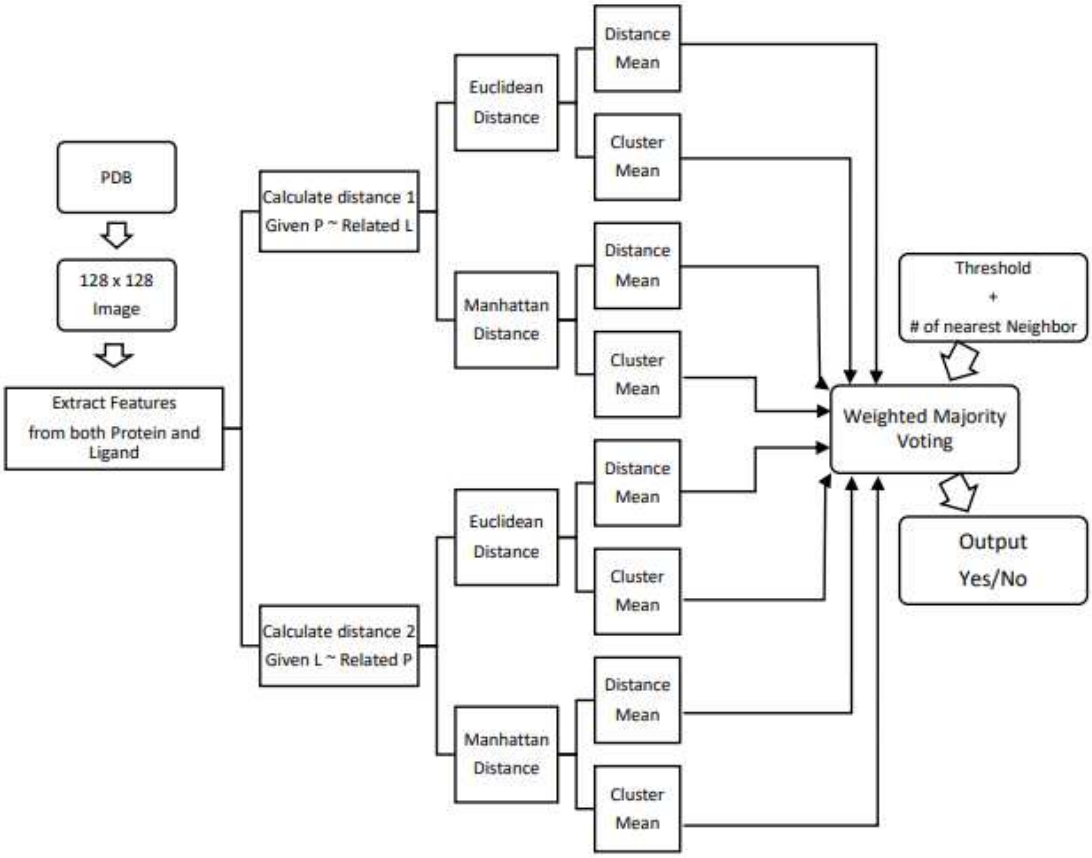

Figure 6. Block Diagram of Similarity Based Clustering.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.27743v1 | CC BY 4.0 Open Access | rec: 19 May 2019, publ: 19 May 2019

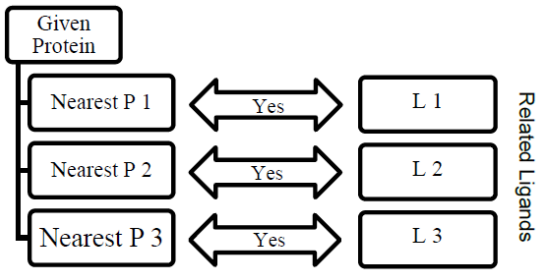

Figure 7. Relation between given protein and related ligands.

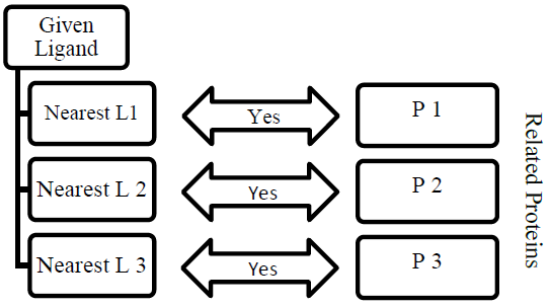

Figure 8. Relation between given ligand and related proteins.

1. **CoMOGrad and PHOG:** CoMOGrad stands for Co-occurrence Matrix of the Oriented Gradient of Distance Matrices and PHOG stands for Pyramid Histogram of Oriented Gradient (Karim et al., 2015). This methodology also uses the α carbon distance matrix of protein. The dimension of all distance matrix is converted to 128×128. In CoMOGrad, the gradient angle and magnitude is computed from the distance matrix and the values are quantized. Quantization is a compressing technique which compresses a range of values to a single quantum value. In this methodology, the values are quantized to 16 bins which produce a co-occurrence matrix which is 16×16 matrix. The matrix is converted into a vector of size 256. Quadtree from the distance matrix is created with the desired level in PHOG. Gradient Oriented Histogram of each node is calculated with the preferred number of bins and bin size. In gradient oriented histogram an image is divided into small sub-images called cells and histogram of edge orientations are accumulated within the cell. The combined histogram entries are used as the feature vector describing the object. Total features which are the multiplication of total nodes and number of bins are incorporated in the vector with the size of the total number of features. The vector is normalized by dividing it with the sum of its components.

2. **Hybrid Local Binary Pattern (Hybrid LBP):** Local Binary Pattern (Ojala et al., 1994) is a procedure of local binary pattern histogram. We have used all the five feature groups described in the last section for structural class prediction problem.

Distance can only be calculated between proteins or between ligands. We've used K-nearest neighbor and Clustering method to calculate these distances.

1. RELATEDLIGANDS($\mathbb{NP}$): For a given Protein, find K-nearest proteins. The ligands those binds with the above nearest proteins, are the Related Ligands for the given protein (See Figure 7).

2. RELATEDPROTEINS($\mathbb{NL}$): For a given Ligand, find K-nearest ligands. The proteins those binds with the above nearest ligands, are the Related Proteins for the given ligand (See Figure 8).

To find the distances between pairs of ligands and proteins are calculated using Euclidean and Manhattan distances. Threshold is the boundary between similarity and dissimilarity in terms of distance. If distance is less than the threshold, then prediction in positive similarity, else the prediction is negative

similarity. Threshold of each category of distances is the average of minimum and maximum distance based on the number of nearest neighbors.

For a given pair of Protein and Ligand, we want to predict if the will bind with each other or not. For measuring distance d_l, from the given protein, we searched for k-nearest proteins and found the k related ligands accordingly. Then we've calculated the distance using above mentioned methods. Then we've taken the vote for the binding class by all categories of distances based their thresholds. Then finally, we've used weighted majority voting mechanism to predict the binding class.

Hyperparameters

There are a number of hyperparameters of our proposed method.

1. **Number of nearest neighbors:** Our algorithm's prediction accuracy is highly dependent on the number of nearest neighbors for finding both $\text{RELATEDLIGANDS}(\mathbb{NP})$ and $\text{RELATEDPROTEINS}(\mathbb{NL})$. We've used 5 nearest neighbors in this experiment.

2. **Threshold:** This is the threshold of distance for determining whether two proteins or two ligands are similar or not. For a higher value of threshold, there is a higher possibility for our algorithm to predict positive binding class for the majority of the Protein-Ligand pairs. And the lower the threshold is, the higher is the possibility of negative binding class prediction. We've taken the average of distances among 5 nearest neighbors as our threshold for each category of the distances.

RESULTS AND DISCUSSION

This section is the description of our experiments performed in this study. Some of the experiments were carried out in a personal desktop computer having Intel Core i3 and 4 GB RAM and others were experimented in a Computing Machine provided by CITS, United International University which was equipped with 8 core processors each having a Dell R 730 Intel Xeon Processor (E5-2630 V3) with 2.4 GHz speed and 18.5 GB memory. Java language was used for data preprocessing including feature generation using OpenCV software library, negative data generation and data merging using Eclipse IDE with Java 8 standard edition. Python language was used to implement our algorithm using the Spyder IDE. Weka tool was used to run the traditional classification algorithms for the comparison with our algorithm. We've used Leave-One-Out validation method to get the accuracy of our model.

Analysis of Features

A different set of parameters were used for each classifiers used in this research. A linear searching was used with no distance weighting for KNN. In case of the Naive Bayesian Classifier, SVM, a polynomial kernel was used with $c = 1.0$ and $\varepsilon = 1.0w^{-2}$. Data was normalized before supplying to the classifier. J48 decision tree classifier was used in Adaboost classifier as the weak base classifier. Classifier number of iterations was set to 100 for Random Forest.

Results in terms of average accuracy in 3-fold cross-validation of protein images are given in Table 3. The highest percentage of correctly classified instances achieved for each of the classifiers are indicated by the boldly faced values of the table.

After running the experiments for our five feature groups ABCDE classifies the highest percentage of correct instances in Random Forest, Adaboost and SVM among all other feature groups. Feature scaled B and D individually provides the highest accuracy in Naive Bayesian and KNN. As the whole combination of all feature groups accuracy gives the highest percentage than any other feature group, thus we conclude that the best performing feature group combination is ABCDE and the best classifier is Random Forest classifier.

Effectiveness in structural class prediction

In this section, we compare the performance of our proposed method with CoMOGrad and PHOG (Karim et al., 2015) along with our previous published literature Wavelet and Pyramid Histogram Features for Image Based Leaf Detection (Ahmed et al., 2019). For comparison with our methodology in this literature, we applied CoMOGrad and Phog techniques and Wavelet and Pyramid Histogram techniques in our dataset of 11052 instances and later applied SMOTE for reducing class imbalance problem. We conducted experiments with different classifiers using the same parameters as we did for feature analysis with the feature groups. The results are given in Table 4. From Table 4 it can be comprehended that our feature

Image Type	Feature Type	Classifiers				
		KNN	Naive Baysian	SVM	Ada Boost	Random Forest
Non Scaled	A	77.70	32.48	68.17	84.11	87.53
Scaled	A	78.55	52.55	79.61	84.22	86.16
Non Scaled	B	74.75	35.65	71.23	83.45	86.85
Scaled	B	76.90	**60.04**	79.26	82.80	84.74
	C	66.96	21.79	44.49	62.26	69.92
Scaled	D	**84.11**	51.23	71.29	83.17	85.05
Scaled	E	83.76	51.24	71.28	83.23	84.94
Non Scaled	AB	76.70	33.53	78.39	85.87	88.43
Non Scaled	ABC	68.27	34.12	82.50	86.74	88.61
Non Scaled + Scaled	ABCD	73.06	35.46	85.47	87.26	89.21
Non Scaled + Scaled	ABCDE	74.72	37.45	**86.12**	**87.74**	**89.49**

Table 3. Classifier accuracies for different types of feature and groups of features.

Feature Type	Classifiers				
	KNN	Naive Baysian	SVM	Ada Boost	Random Forest
Karim et al.Karim et al. (2015)	**87.41**	**59.50**	**87.67**	84.19	85.49
Ahmed et al.Ahmed et al. (2019)	69.36	36.22	67.30	79.92	84.58
this paper	74.72	37.45	86.12	**87.74**	**89.49**

Table 4. Comparison of the proposed features in this paper with Karim et al. (2015) and Ahmed et al. (2019) for structural class prediction.

group ABCDE outperforms CoMOGrad and PHOG in Random Forest and in Adaboost. CoMOGrad and PHOG surpassed our feature groups in KNN, Naive Bayesian and SVM.It can be noted that the combination of our feature groups are three-fourths of CoMOGrad and PHOG. It also can be discerned that the accuracy percentage in Random Forest is higher than all the classifier results. Thus, our novel features can classify more instances than CoMOGrad and PHOG. We have also noticed that our feature groups outperform the features of our previous literatureAhmed et al. (2019) on all classifiers.

We have revealed the precedence of our methodology over CoMOGrad and PHOG (Karim et al., 2015) and Wavelet and Pyramid Histogram Features for Image Based Leaf Detection(Ahmed et al., 2019). The same feature groups were used for leaf detection (Ahmed et al., 2019) with the dataset consisting of RGB images of leaves. Unlike only gray histogram used on this paper, blue, green and red histograms were used to generate features in each feature group and the accuracy result of each classifier was high. The distance matrix of α carbons or the protein images were black and white, thus only gray histogram was used as a feature.

We also used Scale-invariant feature transform(SIFT)(Lowe, 2004) methodologies in our experiments. Each descriptor has a 128-dimensional feature vector. The number of the descriptors of SIFT from every image is not specific so we cannot use traditional machine learning techniques. Hence to apply traditional machine learning procedure and specify the feature vector, we split the image into 16 slices and took one

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.27743v1 | CC BY 4.0 Open Access | rec: 19 May 2019, publ: 19 May 2019

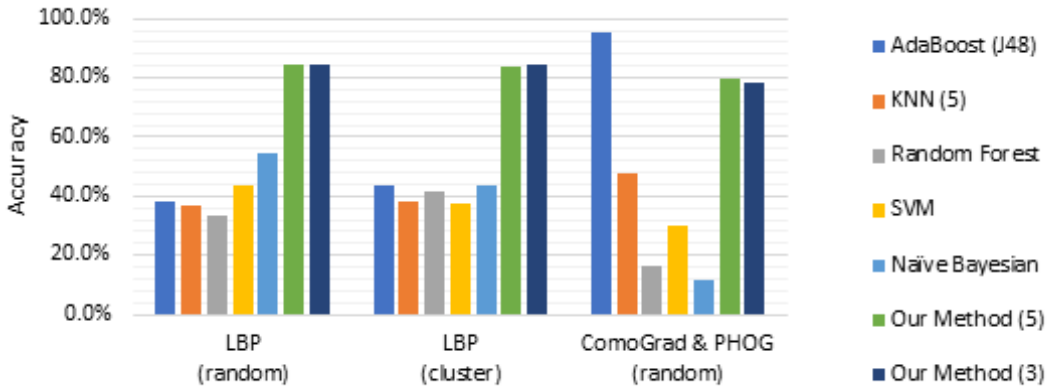

Figure 9. Barplot showing the performance of different algorithms on ligand-binding dataset.

descriptor from each of the slice images. Therefore we got 2048 number of attributes(8x16) from each image. We tested the dataset with the same classifiers mentioned in this paper. The results didn't turn up to be better or close to our proposed methodology in this literature.

Effectiveness in ligand-binding prediction

Sensitivity is the true positive rate regarding the positive instances. As we had to generate the negative data artificially, sensitivity is the actual scale of performance measuring where positive data were the actual data. Using the thresholds gained using the negative data, sensitivity of our algorithm is very good comparing to other existing algorithms shown in Table 5 and Figure 9.

Features	AdaBoost	KNN	Random Forest	SVM	Naive Bayesian	Our Method
LBP (random)	40.00%	43.50%	22.00%	36.80%	45.20%	**91.33%**
LBP (cluster)	51.90%	44.30%	52.20%	49.00%	43.70%	**91.60%**
CoMOGrad & PHOG (random)	**95.20%**	47.60%	16.10%	29.70%	11.30%	79.86%

Table 5. Sensitivity Comparison among different methods for ligand-binding prediction.

We have generated three different datasets based on three different features. Hybrid LBP gives 736 long feature vectors from protein images and 677 long feature vectors from ligand images. So, for one protein-ligand pair we've got 1413 (736+677) attributes and one Binding Class value as one instance. The above mentioned two types of negative data (random and Clustering-Based Undersampling) were generated using Hybrid LBP for balancing the data. CoMOGrad and PHOG gives 1021 or 1020 long feature vectors from protein image, but for ligand images, it gives 1020 long feature vectors. We assumed "0" as the last feature in protein where features were 1020 long, to make it 1021 long feature. So, for one protein-ligand pair we've got 2041 (1021+1020) attributes and one Binding Class value as one instance. Random negative undersampling was used in CoMOGrad and PHOG but Clustering-Based Undersampling was not possible as some clusters couldn't get any unseen pairs of protein and ligand. Our method was used based on 5 and 3 nearest neighbors and shown on the above table and chart.

We can see that AdaBoost works better than our algorithm in terms of sensitivity in ComoGrad and PHOG dataset. Because, Ligand data were so small in terms of number of atoms that ComoGrad and PHOG gave zeros for most of the ligands. But our algorithm's overall performance is better than other machine learning algorithms in the three different feature datasets.

CONCLUSIONS

In this paper, we showed how accurately we can detect protein classes using the combination of different image based feature groups generated from protein images. We also propose a simple similarity-based

clustering method to predict Protein-Ligand Binding without using deep-learning or neural-networks. This simple distance-based algorithm is quite effective compared to complex machine learning algorithms. Our main limitation was the missing negative data. If we had the actual negative data, we could've determined the perfect thresholds for each category of distances, and that would give us more accurate prediction. Another problem was dimensions of small Ligands as we're using image-based features. As the advancement of deep learning, neural network, and many other deep learning techniques are being used to classify images, many remarkably interesting applications can be made. For our future advancement, we wish to introduce new features to improve accuracy, use new tools and explore other fields of computer vision such as human emotion detection. In addition, we will try to extract some unique features from the Ligand dataset so that the dimensionality problem doesn't affect our Protein-Ligand binding prediction.

ACKNOWLEDGMENTS

We thank Rezaul Karim for sharing the SQL dataset files and algorithms to generate Distance Matrix from PDB files for CoMOGrad and PHOG.

REFERENCES

Ahmed, A. A. N., Haque, H. M. F., Rahman, A., Ashraf, M. S., and Shatabda, S. (2019). Wavelet and pyramid histogram features for image-based leaf detection. In Abraham, A., Dutta, P., Mandal, J. K., Bhattacharya, A., and Dutta, S., editors, *Emerging Technologies in Data Mining and Information Security*, pages 269–278, Singapore. Springer Singapore.

Bradski, G. and Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.".

Brady, G. P. and Stouten, P. F. (2000). Fast prediction and visualization of protein binding pockets with pass. *Journal of computer-aided molecular design*, 14(4):383–401.

Chaires, J. B. (2008). Calorimetry and thermodynamics in drug design. *Annu. Rev. Biophys.*, 37:135–151.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Chi, P.-H., Scott, G., and Shyu, C.-R. (2005). A fast protein structure retrieval system using image-based distance matrices and multidimensional index. *International Journal of Software Engineering and Knowledge Engineering*, 15(03):527–545.

Chothia, C. and Lesk, A. M. (1986). The relation between the divergence of sequence and structure in proteins. *The EMBO journal*, 5(4):823–826.

Fox, N. K., Brenner, S. E., and Chandonia, J.-M. (2013). Scope: Structural classification of proteins—extended, integrating scop and astral data and classification of new structures. *Nucleic acids research*, 42(D1):D304–D309.

Holm, L. and Sander, C. (1993). Protein structure comparison by alignment of distance matrices. *Journal of molecular biology*, 233(1):123–138.

Holm, L. and Sander, C. (1997). Dali/fssp classification of three-dimensional protein folds. *Nucleic acids research*, 25(1):231–234.

Jain, A. K. and Bhattacharjee, S. (1992). Text segmentation using gabor filters for automatic document processing. *Machine vision and applications*, 5(3):169–184.

Jain, A. K. and Farrokhnia, F. (1991). Unsupervised texture segmentation using gabor filters. *Pattern recognition*, 24(12):1167–1186.

Karim, R., Aziz, M. M. A., Shatabda, S., Rahman, M. S., Mia, M. A. K., Zaman, F., and Rakin, S. (2015). Comograd and phog: From computer vision to fast and accurate protein tertiary structure retrieval. *Scientific Reports*, 5:13275 EP –. Article.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

Mehrotra, R., Namuduri, K. R., and Ranganathan, N. (1992). Gabor filter-based edge detection. *Pattern recognition*, 25(12):1479–1494.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2012). *Foundations of machine learning*. MIT press.

Ojala, T., Pietikainen, M., and Harwood, D. (1994). Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *Pattern Recognition, 1994. Vol.*

1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on, volume 1, pages 582–585. IEEE.

Ojala, T., Pietikainen, M., and Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987.

Patching, S. G. (2014). Surface plasmon resonance spectroscopy for characterisation of membrane protein–ligand interactions and its potential for drug discovery. *Biochimica et Biophysica Acta (BBA)-Biomembranes*, 1838(1):43–55.

Rossi, A. M. and Taylor, C. W. (2011). Analysis of protein-ligand interactions by fluorescence polarization. *Nature protocols*, 6(3):365.

Shyu, C.-R., Chi, P.-H., Scott, G., and Xu, D. (2004). Proteindbs: a real-time retrieval system for protein structure comparison. *Nucleic Acids Research*, 32(suppl_2):W572–W575.

Singh, A. P. and Brutlag, D. L. (1997). Hierarchical protein structure superposition using both secondary structure and atomic representations. In *Ismb*, volume 5, pages 284–293.

Sousa, S. F., Ribeiro, A. J., Coimbra, J., Neves, R., Martins, S., Moorthy, N., Fernandes, P., and Ramos, M. (2013). Protein-ligand docking in the new millennium–a retrospective of 10 years in the field. *Current medicinal chemistry*, 20(18):2296–2314.

Srivastava, S., Lal, S. B., Mishra, D., Angadi, U., Chaturvedi, K., Rai, S. N., and Rai, A. (2016). An efficient algorithm for protein structure comparison using elastic shape analysis. *Algorithms for Molecular Biology*, 11(1):27.

Steinbrecher, T. and Labahn, A. (2010). Towards accurate free energy calculations in ligand protein-binding studies. *Current medicinal chemistry*, 17(8):767–785.

TAYLOR, W. R. (1999). Protein structure comparison using iterated double dynamic programming. *Protein Science*, 8(3):654–665.