BABEŞ-BOLYAI UNIVERSITY
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Development of deep learning models for complex problems

Summary of the PhD thesis

Keywords: machine learning, bioinformatics, computer vision, deep learning

PhD student: Mihai Teletin
Scientific supervisor: Prof. Dr. Czibula Gabriela

July 2020

Contents

List of Figures **7**

List of Tables **8**

List of publications **11**

Introduction **13**

1 New Machine Learning Models for Intra-Protein Data Analysis **19**
1.1 Using unsupervised learning methods for enhancing protein structure insight 20
1.1.1 Methodology . . . 20
1.1.2 Results and discussion . . . 21
1.1.3 Conclusions and further work . . . 22
1.2 Deep autoencoders for additional insight into protein dynamics . . . 22
1.2.1 Methodology . . . 23
1.2.2 Results and discussion . . . 24
1.2.3 Conclusions and further work . . . 27

2 New Machine Learning Models for Inter-Protein Data Analysis **29**
2.1 Using clustering models for uncovering proteins' structural similarity . . . 30
2.1.1 Methodology . . . 30
2.1.2 Experimental results and discussion . . . 32
2.1.3 Conclusions and future work . . . 33
2.2 *AutoSimP*: An approach for predicting proteins' structural similarities using an ensemble of deep autoencoders . . . 33
2.2.1 Methodology . . . 33
2.2.2 Results and discussion . . . 34
2.2.3 Conclusions and future work . . . 35
2.3 *AnomalP*: An approach for detecting anomalous protein conformations using deep autoencoders . . . 35
2.3.1 Methodology . . . 36
2.3.2 Experimental evaluation . . . 38
2.3.3 Discussion . . . 38
2.3.4 Conclusions and future work . . . 39

3 Deep Learning Models in Computer Vision **41**
3.1 Detecting false signatures using Convolutional Neural Networks . . . 42
3.1.1 The proposed approach . . . 42
3.1.2 Results and discussion . . . 43
3.1.3 Conclusions and further work . . . 44
3.2 Lightweight deep learning models for fruits recognition . . . 44
3.2.1 Our approach . . . 44

3.2.2 Experimental evaluation 45
3.2.3 Conclusions and future work 47
3.3 A document detection technique using convolutional neural networks for optical character recognition systems 47
3.3.1 The proposed approach 48
3.3.2 Experimental evaluation 48
3.3.3 Conclusions and further work 50
3.4 *CVSimP*: An approach for predicting proteins' structural similarity using one-shot learning 50
3.4.1 Methodology 50
3.4.2 Experimental results and discussion 51
3.4.3 Conclusions and future work 52

Final Conclusions and Future Work **53**

Contents of the thesis

List of Figures **4**

List of Tables **6**

List of publications **7**

Introduction **9**

1 Background **15**
1.1 Proteins and conformational transitions 15
1.1.1 Structural alphabets 16
1.1.2 Literature review on protein data analysis 18
1.2 Computer vision 22
1.2.1 Projective transformations 23
1.2.2 Related work 23
1.3 Machine learning 25
1.3.1 Supervised learning 25
1.3.2 Unsupervised learning 29
1.4 Conclusions 30

2 New Machine Learning Models for Intra-Protein Data Analysis **31**
2.1 Using unsupervised learning methods for enhancing protein structure insight 32
2.1.1 Methodology 32
2.1.2 Results and discussion 34
2.1.3 Conclusions and further work 38
2.2 Deep autoencoders for additional insight into protein dynamics 39
2.2.1 Methodology 39
2.2.2 Results and discussion 41
2.2.3 Conclusions and further work 46

3 New Machine Learning Models for Inter-Protein Data Analysis **48**
3.1 Using clustering models for uncovering proteins' structural similarity 49
3.1.1 Methodology 50
3.1.2 Experimental results and discussion 51
3.1.3 Conclusions and future work 55
3.2 AutoSimP: An approach for predicting proteins' structural similarities using an ensemble of deep autoencoders 55
3.2.1 Methodology 56
3.2.2 Results and discussion 58
3.2.3 Conclusions and future work 61
3.3 AnomalP: An approach for detecting anomalous protein conformations using deep autoencoders 61

3.3.1 Methodology 63
3.3.2 Experimental evaluation 68
3.3.3 Discussion 70
3.3.4 Conclusions and future work 75

4 Deep Learning Models in Computer Vision **76**
4.1 Detecting false signatures using Convolutional Neural Networks 77
4.1.1 Problem relevance and difficulty 78
4.1.2 The proposed approach 78
4.1.3 Results and discussion 80
4.1.4 Conclusions and further work 82
4.2 Lightweight deep learning models for fruits recognition 82
4.2.1 Problem relevance and difficulty 83
4.2.2 Our approach 83
4.2.3 Experimental evaluation 85
4.2.4 Conclusions and future work 88
4.3 A document detection technique using convolutional neural networks for optical character recognition systems 88
4.3.1 The proposed approach 89
4.3.2 Experimental evaluation 91
4.3.3 Conclusions and further work 93
4.4 CVSimP: An approach for predicting proteins' structural similarity using one-shot learning 93
4.4.1 Methodology 94
4.4.2 Experimental results and discussion 97
4.4.3 Conclusions and future work 99

Final Conclusions and Future Work **101**

Bibliography **105**

List of Figures

1.1 HAC results 21
1.2 HAC 1GO1 and 1L3P 22
1.3 Autoencoder visualization of 1JT8 25
1.4 Autoencoder visualization of 1P1L 25
1.5 Comparative average similarities for 1JT8 26
1.6 Comparative average similarities for 1P1L 27

2.1 Overview of *AutoSimP* approach. 34
2.2 Overview of *AnomalP* approach. 37
2.3 AUC values obtained by *AnomalP* using the *rank-based* and the *angles-based* representation for the conformations. The error bars represent 95% CIs. . 39

3.1 Evolution of validation accuracy for a particular experiment run on MobileNet V2 that is trained from scratch. 45
3.2 Test accuracies of the 10 experiments obtained by MobileNet V2. 46
3.3 Evolution of validation accuracy for a particular experiment run on MobileNet V2 that is trained using ImageNet initialization and data augmentation. 47
3.4 Test samples including the original image and the projection of the detected receipt. The prediction is marked by red dots and the ground truth is marked by green ones. It is worth mentioning that the model correctly marked the cash receipt even though the corner was obstructed. 49

List of Tables

1.1 Clustering results. 21
1.2 *IntraPS* for proteins 1JT8 and 1P1L, using the two considered representations. 26
1.3 *InterPS* for proteins 1JT8 and 1P1L, using the original and encoded data. . 27

2.1 Clustering results. 32
2.2 Experimental results. 35

3.1 Comparison to related work based on the accuracy evaluation measure. . . 44
3.2 Our test results for various settings. 95% CIs are used. 46
3.3 Comparison to related work in terms of test accuracy 47
3.4 Results with 95% CIs. 49
3.5 Comparison to related work based on Hough values. 95% CIs are provided for our models. 50
3.6 Proteins considered for this study [ATC18]. 52
3.7 Experimental results obtained by applying $CVSimP$. A comparison to previous work is depicted 52

List of publications

All rankings are listed according to the classification of journals[1] and conferences[2] in Computer Science.

Publications in Web of Science - Science Citation Index Expanded

[CCT19] Gabriela Czibula, Carmina Codre, and **Mihai Teletin**. *AnomalP: A new approach for detecting anomalous protein conformations using deep autoencoders.* Expert Systems with Applications, 2019, under review (**IF=4.292**)

[ATC18] Silvana Albert, **Mihai Teletin**, and Gabriela Czibula. *Analysing protein data using unsupervised learning techniques.* International Journal of Innovative Computing, Information and Control (IJICIC), pp. 861-880, Volume 14, Number 3, June 2018. (**IF=1.667**)

Rank C, 2 points.

Publications in Web of Science - Conference Proceedings Citation Index

[TC20] **Mihai Teletin** and Gabriela Czibula. *CVSimP: An approach for predicting proteins' structural similarity using one-shot learning.* IEEE 14th International Symposium on Applied Computational Intelligence and Informatics, SACI 2020, Timisoara, in press

Rank C, 2 points.

[DT19] Lorand Dobai and **Mihai Teletin**. *A document detection technique using convolutional neural networks for optical character recognition systems.*European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, ESANN, pp. 547–552, 2019.

Rank B, 4 points.

[TCC19] **Mihai Teletin**, Gabriela Czibula, and Carmina Codre. *AutoSimP: An approach for predicting proteins' structural similarities using an ensemble of deep autoencoders.*The 12th International Conference on Knowledge Science, Engineering and Management (KSEM 2019), LNAI 11776, pp. 4954, 2019.

Rank B, 4 points.

[TCAB18] **Mihai Teletin**, Gabriela Czibula, Silvana Albert, and Mariana-Iuliana Bocicor. *Using unsupervised learning methods for enhancing protein structure insight.* International Conference on Knowledge Based and Intelligent Information and Engineering

[1] https://uefiscdi.ro/premierea-rezultatelor-cercetarii-articole
[2] http://portal.core.edu.au/conf-ranks/

Systems, KES 2018, Belgrade, Serbia, Procedia Computer Science, 126, pp. 126-135, 2018.

Rank B, 2 points.

[TCB+18] **Mihai Teletin**, Gabriela Czibula, Mariana-Iuliana Bocicor, Silvana Albert, and Alessandro Pandini. *Deep autoencoders for additional insight into protein dynamics.* International Conference on Artificial Neural Networks (ICANN), Rhodes, Greece, LNCS, volume 11140, pp. 78-89, 2018.

Rank B, 1.33 points.

[TCB19] **Mihai Teletin**, Gabriela Czibula, and Maria-Iuliana Bocicor. *Using clustering models for uncovering proteins' structural similarity.* IEEE 13th International Symposium on Applied Computational Intelligence and Informatics, SACI 2019, Timisoara, Romania, pp. 185-190, 2019

Rank C, 2 points.

[TD19] **Mihai Teletin** and Lorand Dobai. *Lightweight models for fruits recognition.* IEEE 13th International Symposium on Applied Computational Intelligence and Informatics, SACI 2019, Timisoara, Romania, pp. 69-74, 2019.

Rank C, 2 points.

[ACT18] Silvana Albert, Gabriela Czibula, and **Mihai Teletin**. *Analyzing the impact of protein representation on mining structural patterns from protein data.* IEEE 12th International Symposium on Applied Computational Intelligence and Informatics, SACI 2018, Timisoara, Romania, pp. 533-538, 2018.

Rank C, 2 points.

[ICT17] Vlad-Sebastian Ionescu, Gabriela Czibula, and **Mihai Teletin**. *Supervised learning techniques for body mass estimation in bioarchaeology.* IEEE 7th International Workshop on Soft Computing Applications (SOFA), Springer, pp. 71-86, 2017.

Rank C, 2 points.

[ITV16] Vlad-Sebastian Ionescu, **Mihai Teletin**, and Estera-Maria Voiculescu. *Machine learning techniques for age at death estimation from long bone lengths.* IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI 2016), Timisoara, Romania, pp. 457 - 462, 2016.

Rank C, 2 points.

Publications in journals and conference proceedings indexed in international databases

[Tel17] **Mihai Teletin**. *Machine Learning Techniques for Detecting False Signatures.* Studia Universitatis Babes-Bolyai, Informatica 62(1), pp. 49–59, 2017. (**indexed Mathematical Reviews**)

Rank D, 1 point.

[BPC+17] Maria-Iuliana Bocicor, Alessandro Pandini, Gabriela Czibula, Silvana Albert, and **Mihai Teletin**. *Using computational intelligence models for additional insight into protein structure.* Studia Universitatis Babes-Bolyai, Informatica 62(1), pp. 107–119, 2017. (**indexed Mathematical Reviews**)

Rank D, 0.33 points.

Publications score: 26.66 points.

Introduction

The main domain of study that is pursued in our doctoral thesis is Machine Learning (ML). The PhD Thesis is entitled "Contributions in Developing Deep Learning Models for Complex Problems". Our work is focused on developing new *Deep learning* models for solving complex problems from two domains namely *bioinformatics* and *computer vision*.

Machine Learning (ML) is the main research direction in Artificial Intelligence (AI) domain. The goal of this discipline is to develop models that make predictions and that are capable to improve these predictions by gaining experience. Nowadays, the applicability of ML in computer vision and bioinformatics is a very popular topic. Therefore a significant part of the research from these two particular domains is oriented towards ML.

Our research in *applied machine learning* started with developing Artificial neural network models (ANN) for two important tasks in computational archaeology: *body mass* and *age at death* estimation from human skeletal remains. Both problems are of major importance in paleontological and archaeological research, since they can provide useful information about past populations, such as their healthiness, different social aspects, the influence of environmental factors and others. The task of estimating the body mass from human skeletal remains based on bone measurements was investigated from a *machine learning* perspective in [ICT17]. Two supervised learning-based regression models were proposed, using *artificial neural networks* and *support vector machines*, for expressing good nonlinear mappings between skeletal measurements and body mass. Several experiments performed on an open source dataset showed that the proposed applications of machine learning-based algorithms lead to better results than the current state of the art. In [ITV16] we applied *artificial neural networks* and *support vector machines* for estimating age at death of cadavers and proved that they outperform existing mathematical approaches on a number of case studies derived from publicly available data used for this task.

An important subdiscipline of ML that has emerged from the study of Artificial neural networks is *Deep learning*. The field has recorded notable success in various fields while it managed to improve state of the art performance. The potential of Machine Learning methods was successfully leveraged for tackling problems that were once considered very challenging. For instance, one of the most complex problems from the computer vision domain, the ImageNet challenge [DDS+09] was solved using a convolutional neural network [KSH12, SVI+15, Cho16]. On the other hand, a complex problem in the bioinformatics field is protein analysis. Various work has emphasized the fact that Machine learning techniques are suitable for tackling this problem [LJN06, BDF+15].

Approached problems. Motivation

The thesis is focused on developing Machine learning solutions for two distinct real life domains, namely *computer vision* and *bioinformatics*. At the first sight these two domains seem to be different. However, one can argue that both fields include problems that are challenging and difficult to solve. Moreover, from a computational perspective the main connection between the real life problems from these two fields of study is their *complexity*.

The *complexity* of the problems approached in the current thesis may be understood from several viewpoints. First, a side of the complexity of a problem may be related to its *difficulty* which make the problem impossible to solve with traditional programming, i.e. classical algorithms and methods. Moreover we consider *complexity* of a problem also linked to the input data which has to be processed: in most cases a large amount of (possibly unlabeled) data has to be analyzed; in certain cases the data contains noise. For these reasons, we consider Machine learning methods very well suited for handling the *complexity* of the problems, as previously discussed. On the other hand, from a computational perspective we refer to the complexity of a problem as connected to its *NP-completeness* or *NP-hardness*. NP-complete problems are problems whose solutions can be verified in polynomial time while for NP-hard problems this property may not stand. In general, both classes of problems are not solvable in polynomial time [BCB94]. From a computer vision perspective there are some results in the literature that show that some basic problems such as image matching are *NP-complete* [KU03]. On the *bioinformatics* domain, there are some preliminary results that suggest that protein folding is an *NP-complete* problem [BL98, BBCG13].

Focusing more on our particular field of studies, one can argue that it is really hard to design a solution for problems from *bioinformatics* and *computer vision* using classical algorithms. Moreover, it is also well known that meta and hyper heuristics can't achieve state of the art performance. On the other hand, the capability of Machine learning techniques to adapt to unknown situations through learning may be useful. In fact, this capability motivates us to use such techniques for tackling the two fields of study by proposing new techniques that are capable to achieve state of the art performance.

We plan to focus our research towards developing Machine learning models for *protein analysis* and *computer vision* tasks. As previously shown, the main connection between the two fields that we're studying is their complexity. We believe that this is the main reason that Deep learning techniques may be very effective for solving problems related to these fields.

Protein analysis is of great interest both in the computational biology and bioinformatics research. Proteins are large, complex molecules with crucial roles in the functioning of living organisms. Understanding the underlying mechanisms by which proteins achieve their structures and substructures, as well as those involved in the conformational transitions may contribute to a deeper comprehension of the involved biological processes. Although the stable 3D structure of a protein is defined by a unique topology (i.e. fold), this structure is not static and it is now widely accepted that proteins are dynamic objects [TT09]. According to various external factors from the protein's environment (e.g. temperature, interaction with other molecules), modifications in proteins' structures occur during their biological functions. A protein will thus acquire a limited number of conformations and will have the ability to transition between alternative conformations. Understanding protein dynamics and how these conformational transitions occur is essential for the comprehension of biomolecular interactions, which is of paramount importance in the process of developing new drugs that can inhibit proteins' uncontrolled behaviour [MMC08].

Both the importance and the complexity of the problem motivate us to investigate the usefulness of *machine learning* models and methods for the analyzing and detecting the conformational changes in proteins. Because of the protein dynamics speed and complexity, analyzing a protein for a relatively short period of time can be used in order to extract large volume of data. It is well known that such mass amount of data can be successfully exploited using various Machine learning techniques.

In the *protein analysis* field we're going to focus our research towards developing Machine learning methodologies for analyzing protein structure with the purpose to help us understand and analyze the complexity of the proteins: use unsupervised learning techniques for inter and intra protein structure analysis; develop efficient encoding techniques; using

autoencoders for supervisedly classifying proteins in superfamilies and detecting anomalous protein conformations.

Secondly, we aim to develop Deep learning techniques for some *computer vision* problems: improve performance on a classification task, namely signature classification; demonstrate the extent of convolutional neural networks for image preprocessing by developing document localization and deskewing techniques.

Thesis structure

The rest of the thesis is structured as follows. The first bioinformatics related research direction is intra protein analysis and it's going to be covered in Chapter 1. The experiments conducted are illustrated in Sections 1.2 and 1.1. More exactly, a *clustering* based experiment will be depicted in section 1.1. From a methodology perspective, we present the clustering techniques used, the evaluation measures, some visualization techniques and the protein encoding process in Section 1.1.1. Moreover, the experimental results and discussions are depicted in Section 1.1.2. An autoencoder based experiment is presented in 1.2. The methodology is introduced in Section 1.2.1 and covers protein representation details and autoencoder parameter settings. The experiments and the obtained results are discussed in Section 1.2.2.

Chapter 2 reviews our results related to inter proteins analysis. Therefore, Section 2.1 contains a clustering based experiment that manages to outperform previous work on the same topic. Three representation for the proteins' conformational transitions are proposed in Section 2.1.1. The experiments are performed using all these representations and the obtained results are depicted and compared in Section 2.1.2. The method is then extended from a different point of view in Section 2.2. More exactly, an ensemble of autoencoders is developed for tackling the same problem. From the perspective of the approach, we are describing the data representation and the model in Section 2.2.1. The experimental results are then highlighted in Section 2.2.2. The method presented in this section managed to achieve state of the art performance. An approach named *AnomalP* based on *deep autoencoders* for detecting anomalous conformational transitions of proteins is introduced in Section 2.3. The method is based on an ensemble of autoencoders and is described in Section 2.3.1. The experiments and the obtained experimental results are presented in Section 2.3.2, while Section 2.3.3 provides an analysis of the results, as well as a comparison to existing similar work. Section 2.3.4 summarizes the conclusions of the subsection and directions for further improvements and extensions.

Finally, in Chapter 3 we discuss three practical computer vision tasks. Firstly, a signature verification pipeline is introduced in Section 3.1. Then, a solution for an open source dataset for fruits recognition is presented in Section 3.2. The method proposed in Section 3.2.1 is based on lightweight Deep learning models and achieves state of the art performance for both execution time and classification performance as described in 3.2.2. Finally, a *convolutional neural network* based localization technique will be revised in Section 3.3. We study the localization problem while trying to improve state of the art performance for a practical task, document detection and deskewing. An important part of an optical character recognition pipeline is the preprocessing step, whose purpose is to enhance the conditions under which the text extraction is later performed. The results depicted in Section 3.3.2 outperform existing work and can be efficiently used for improving optical character recognition accuracy.

Original contributions

This thesis is focused on two main research directions, computer vision and proteins analysis. Therefore, the original contributions of this thesis are twofold.

1. From a bioinformatics perspective we designed and implemented new Machine learning based methods for performing protein analysis.

2. From a computer vision perspective we developed Deep learning based solutions for some image processing tasks.

The bioinformatics related contributions were included in Chapters 1 and 2 and were published in seven research papers [BPC+17, ATC18, ACT18, TCB+18, TCAB18, TCC19, TCB19]:

- We experimentally highlighted that the information obtained through analyzing proteins conformational transitions capture the relationships between related proteins, relations which are confirmed from a biological perspective [BPC+17]. We investigated a new Machine learning perspective for analyzing protein conformational transitions and proposed a new formalization for the discussed problem. This study represented the starting point of our research.

- We investigated the usefulness of *self organizing maps* and *fuzzy self organizing maps* in identifying the structural relationship between proteins [ATC18]. The computational experiments that we performed on several protein datasets depicted the effectiveness unsupervised learning models to capture the similarity between proteins. The reported results also revealed that *fuzzy* models are able to increase the unsupervised model's performance.

- The impact of RSA values while using *self organizing maps* for studying the internal structure of proteins was analyzed [ACT18]. Using two distinct representations, two case studies were performed for emphasizing the effectiveness of the *self organizing maps* based approach.

- We studied the capability of autoencoders to preserve and illustrate protein information while analyzing protein data [TCB+18]. The Deep learning models were being explored in order to highlight their ability to learn relevant biological patterns, such as structural characteristics. The study was aimed to provide a better comprehension of how protein conformational transitions are evolving in time, within the larger framework of automatically detecting functional motions.

- The extent of clustering as an unsupervised classification method was demonstrated for investigating the relevance of RSA values for predicting internal transitions of proteins [TCAB18]. With the main goal of studying the evolution of RSA values between conformational transitions, we experimentally showed that RSA values are slowly modifying as the protein undergoes conformational changes

- We extended the clustering approach for inter protein analysis [TCB19]. We investigated three representations for a protein based on the probability distributions of certain structural elements within conformational transitions and apply clustering methods to unsupervisedly classify proteins based on their structural similarity. Experiments were performed on two protein datasets. The comparative results revealed that in many cases our proposal performs better than an earlier work in this topic.

- A system based on an ensemble of autoencoders for proteins classification was developed [TCC19]. The goal of the system is to predict the similarity class of a certain protein, considering the similarity class predicted for its conformational transitions. Experiments were performed on real protein data and revealed the effectiveness of our proposal compared with similar existing approaches.

The computer vision related contributions are contained in Chapter 3 and were introduced in three research papers [Tel17, TD19, DT19]:

- We applied Deep learning methods for signature authenticity determination [Tel17]. The possibility to use a supervised learning techniques in order to build models capable to accurately perform such an analysis was investigated. The results reported during the testing phase were encouraging for further work.

- A document detection and deskewing method was designed. It was based on a convolutional neural network and perspective transformation [DT19]. Our work intended to serve as a preprocessing step for an optical recognition system. The main challenge was to improve performance especially on frames which were skewed (slightly rotated) or had cluttered backgrounds. The proposed method achieved good document detection and deskewing results on a dataset of photos of cash receipts.

- A state of the art solution for an open source image classification dataset was proposed [TD19]. The solution was based exclusively on lightweight deep neural networks. We achieved state of the art performance in terms of both classification accuracy and execution speed.

Our perspectives on the problem are new, to the best of our knowledge they have not been investigated in the literature, yet. We are confident that *machine learning* based solutions are applicable in the domains approached in the thesis, *bioinformatics* and *computer vision* and may lead to interesting and valuable information, due to these models' ability to discover hidden patterns in data.

***

The author of the thesis thank lecturer Alessandro Pandini from Brunel University, London for providing the protein data sets used in the experiments performend in Chapters 1 and 2.

Chapter 1

New Machine Learning Models for Intra-Protein Data Analysis

Proteins are the building blocks of any living organism. They are complex macromolecules which contribute to maintaining cellular environments and thus have fundamental roles in biological processes. They represent the end result of the *DNA* decoding process. However, understanding their genesis and folding is still a missing piece from the biology literature. In this Chapter we present our investigation towards applying *unsupervised learning* methods for analyzing internal protein structure and the transitions that they take part into. The main goal is to extract meaningful information about the structural similarity of proteins. The experiments performed on different protein datasets emphasize the effectiveness of unsupervised learning models for capturing the similarity between proteins' structures.

In this Chapter we're highlighting two main original contributions that were published in original papers [TCAB18, TCB+18]:

- In Section 1.1 we use *clustering* as an unsupervised classification method in order to study the relevance of the residues' RSA values for analyzing protein internal transitions [TCAB18]. With the main goal of studying the evolution of RSA values between conformational transitions, we experimentally show that RSA values are slowly changing as the protein suffers conformational changes. The study is aimed to provide a better apprehension of how proteins' conformational transitions are evolving in time, with the broader goal of better understanding protein internal dynamics.

- In Section 1.2, we investigate the usefulness of unsupervised Machine learning methods for uncovering relevant information about protein functional dynamics [TCB+18]. *Autoencoders* are being explored in order to highlight their ability to learn relevant biological patterns, such as structural characteristics. This study is aimed to provide a better comprehension of how protein conformational transitions are evolving in time, within the larger framework of automatically detecting functional motions.

Section 1.1 is structured as follows. We highlight the clustering techniques used, the evaluation measures, some visualization techniques and the protein encoding process in Section 1.1.1. The results of the experiments and some comparisons are depicted in Section 1.1.2. Finally, the conclusions are drawn in Section 1.1.3.

Section 1.2 is structured as follows. The methodology of the approach is discussed in Section 1.2.1 and covers details about protein representation and the parameter settings of the model. Then, the experiments and the obtained results are discussed in Section 1.2.2. Finally, we briefly present the conclusions of the research direction in Section 1.2.3.

1.1 Using unsupervised learning methods for enhancing protein structure insight

In the current section, we illustrate various clustering based experiments for analyzing internal proteins' transitions. The experiments depicted in this section were introduced in the original paper [TCAB18].

In this Chapter we are investigating the usefulness of *clustering* models to unsupervisedly uncover relevant information which would offer a better understanding of proteins' structure, with the broader goal of learning to predict how proteins are evolving in time. In our study we use an internal representation for a protein based on the *relative solvent accessibility* (RSA) values of amino acid residues, considering a substantial set of conformational transitions.

Through several experiments performed on *four* proteins we aim to obtain an empirical evidence that (1) there are some patterns in the way a protein transitions from one conformation to another and that (2) protein conformations which are temporally close are similar to each other. Our experiments show that the RSA values are slowly changing between the proteins' transitions. To the best of our knowledge, a study similar to ours has not been performed, so far, for intra-protein analysis.

1.1.1 Methodology

The experimental methodology used for assessing the effectiveness of clustering in analyzing proteins conformational transitions is further detailed. We conduct a clustering experiment in order to find out how well different conformations of a protein (represented as RSA vectors) cluster together. Moreover, we are going to use PCA analysis in order to create two dimensional visualization of our data. The scikit-learn implementation [PVG+11] was used for clustering, as well as for PCA.

1.1.1.1 Protein datasets

In our experiments *four* proteins for which the *relative solvent accessibility* (RSA) values are available will be used. The selected proteins are: **1GO1** (classified as a Ribosomal Protein), **1JT8** (translation initiation factor classified in the Translation category), **1L3P** (pollen allergen classified as Allergen) and **1P1L** (Periplasmic divalent cation tolerance protein classified as Structural Genomics Unknown Function) [BWF+00].

The dataset for each protein consists of 10000 conformational transitions and it represents the evolution of the protein's structure (conformational transitions) in very small intervals of time. In our computational approach, each conformational transition is represented as a sequence of 99 numerical values, representing the *relative solvent accessibility* (RSA) values of the amino acid residues from the protein's primary structure.

Before applying the unsupervised learning methods, a standard deviation based normalization was used for each of the protein datasets.

1.1.1.2 Clustering

On each of the preprocessed protein datasets, an experiment is first performed using K-means and *hierarchical agglomerative clustering* (HAC) as clustering methods. Having 10000 successive conformations represented by RSA vectors we group them in classes by using a hyperparameter named **step** (s) representing the cardinality of each class. The step induces a new hyperparameter: the number of clusters to be used for the clustering process (K). For example a step $s = 2500$ would mean that we are expecting $K = 4$ classes, namely:

Class 1 - conformations from 1 to 2500; **Class 2** - conformations from 2501 to 5000; **Class 3** - conformations from 5001 to 7500; **Class 4** - conformations from 7501 to 10000.

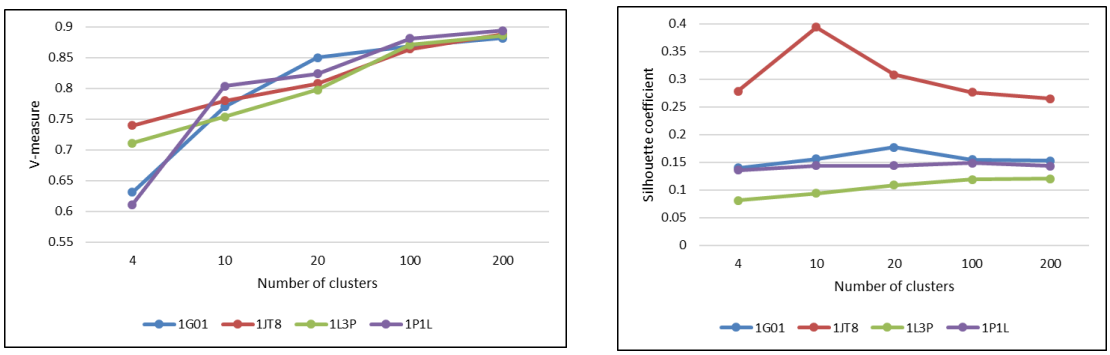
1.1.2 Results and discussion

In this section we present the results of our experiments, as well as a discussion regarding the obtained results from a computational and biological viewpoint. The results obtained by the clustering experiments will be presented, using the experimental methodology described in Section 1.1.1. We have employed both hierarchical agglomerative clustering and K-means for the process described in Section 1.1.1.2. The obtained results are depicted in Table 1.1, where, for each protein, the best values obtained for *V-measure* and *silhouette coefficient* are highlighted. The obtained values for *V-measure* are good enough, ranging between 0.6 and 0.9 which means that classification is performed relatively well. The best values for *V-measure* and *silhouette coefficient* are highlighted.

Protein	**Step**	**K**	**Clustering method**	*V measure*	*Silhouette coefficient*	**Protein**	**Step**	**K**	**Clustering method**	*V measure*	*Silhouette coefficient*
1GO1	2500	4	$K-means$	0.715	0.156	1L3P	2500	4	$K-means$	0.708	0.084
			HAC	0.631	0.140				HAC	0.711	0.081
	1000	10	$K-means$	0.812	0.152		1000	10	$K-means$	0.767	0.101
			HAC	0.770	0.156				HAC	0.754	0.094
	500	20	$K-means$	0.864	**0.179**		500	20	$K-means$	0.802	**0.112**
			HAC	0.850	0.177				HAC	0.798	0.109
	100	100	$K-means$	0.869	0.160		100	100	$K-means$	0.856	0.121
			HAC	0.869	0.155				HAC	0.871	0.119
	50	200	$K-means$	0.875	0.149		50	200	$K-means$	0.880	0.117
			HAC	**0.882**	0.153				HAC	**0.886**	0.120
1JT8	2500	4	$K-means$	0.662	0.282	1P1L	2500	4	$K-means$	0.607	0.145
			HAC	0.740	0.278				HAC	0.611	0.136
	1000	10	$K-means$	0.754	0.321		1000	10	$K-means$	0.774	0.148
			HAC	0.780	0.304				HAC	0.804	0.144
	500	20	$K-means$	0.790	**0.317**		500	20	$K-means$	0.820	0.144
			HAC	0.808	0.308				HAC	0.824	0.144
	100	100	$K-means$	0.861	0.275		100	100	$K-means$	0.869	**0.151**
			HAC	0.864	0.276				HAC	0.881	0.149
	50	200	$K-means$	0.882	0.266		50	200	$K-means$	0.885	0.145
			HAC	**0.888**	0.265				HAC	**0.894**	0.143

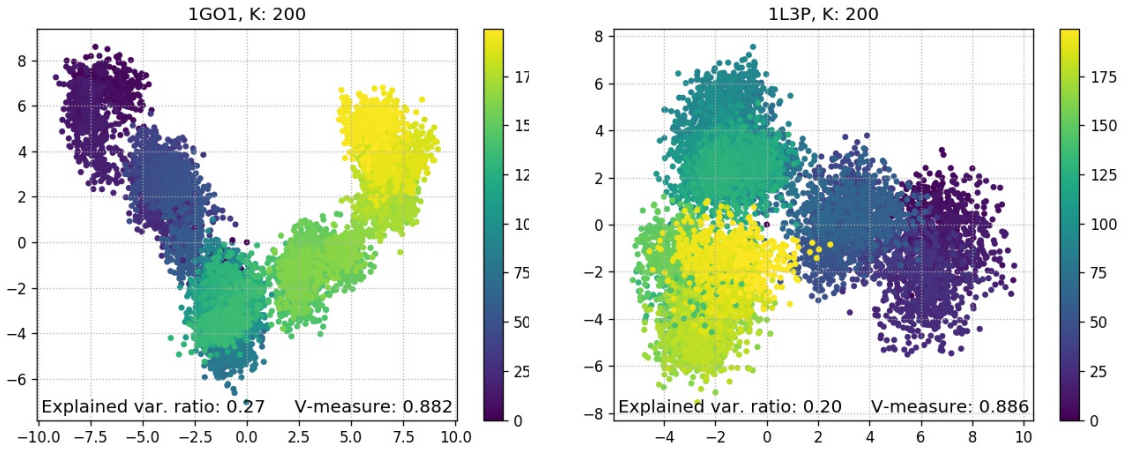
Table 1.1: Clustering results.

From Table 1.1 we observe that, generally, the clusters provided by the hierarchical agglomerative clustering are better than those reported by the partitional K-means method, i.e. HAC provides higher *V-measure* and *silhouette coefficients* for the obtained clusters.



(a) *V-measure.* (b) *Silhouette coefficient.*

Figure 1.1: Results for HAC clustering and different number of clusters, for each protein.



(a) Protein 1GO1.

(b) Protein 1L3P.

Figure 1.2: HAC visualization of proteins 1GO1 (left side) and 1L3P (right side) using PCA.

As the protein undergoes conformational changes, certain parts of its structure are subjected to minor modifications, which are reflected in the positions of the amino acid residues and consequently, in their RSA values. Thus, consecutive conformations are fairly similar from the perspective of their considered representations (RSA values).

To the best of our knowledge, the study presented in this thesis is new in the literature related to RSA driven protein data analysis. The literature review we performed revealed that *unsupervised learning* was applied mainly for analyzing the interaction between proteins, and not for intra-protein analysis, as in our thesis. Most of existing approaches in the literature related to intra-protein analysis use *supervised* or *semi-supervised* learning for classifying proteins according to their structural characteristics. Only few approaches are using *unsupervised learning* methods for analyzing protein conformational transitions.

1.1.3 Conclusions and further work

In this chapter we presented a study towards applying *clustering* as an *unsupervised classification* method for investigating the relevance of RSA values for predicting internal transitions of proteins. *Principal component analysis* was also explored for data visualization and used to examine how RSA values evolve between conformational transitions. The experiments conducted on several proteins highlighted that RSA values are smoothly changing between conformational transitions.

We plan to extend the analysis of the computational results obtained in this chapter from a biological viewpoint. Based on the study previously performed and on previous investigations regarding protein data analysis, we aim to advance our research towards predicting protein conformational transitions.

1.2 Deep autoencoders for additional insight into protein dynamics

In the current section, we introduce various autoencoder based experiments for analyzing the structure of proteins. The experiments depicted in this section were introduced in the original paper [TCB+18].

The contribution of the work presented in this Chapter is twofold. Our first main goal is to investigate the capability of unsupervised learning models. Secondly, we propose two internal representations for a protein with the aim of analyzing which of them is more informative and would drive an autoencoder to better learn structural relationships between proteins. In order to achieve our goals, experiments will be conducted on ensemble of conformations for two proteins, *1JT8* and *1P1L*, using different representations.

To sum up, in this work three research questions are investigated:

RQ1 What is the ability of *autoencoders* to learn in an unsupervised manner the structure of proteins and how does the internal representation for a protein impact the learning process?

RQ2 To what extent is an autoencoder capable to preserve the structural similarity between proteins?

RQ3 Are autoencoders able to express patterns which would be relevant from a biological perspective? More exactly, to what extent are our computational results correlated with the biological perspective?

1.2.1 Methodology

In this section we present the experimental methodology used in supporting our assumption that *autoencoders* can capture, from a computational viewpoint, biologically relevant patterns regarding structural conformational changes of proteins.

In order to answer the first two research questions previously formulated, the experiments will be conducted in two directions.

Firstly, we investigate in Section 1.2.2.2 the ability of an *autoencoder* to preserve the structure of a protein. Two types of representations will be considered in order to identify the one that is best suited for the analysis we are conducting. These representations will be detailed in Section 1.2.1.1.

Secondly, we turn our attention to proteins' structural relationships and we analyze whether self-supervised learning techniques such as autoencoders are able to detect certain underlying patterns within the data. To this end, we conduct a case study on two structurally similar proteins, study that will further be described in Section 1.2.2.3.

1.2.1.1 Protein representations

A protein is a macromolecule with a very flexible and dynamic innate structure [MJC02] that changes shape due to both external changes from its environment and internal molecular forces. The resulting shape is a different conformation. For each conformation of a protein, two different representations of the local geometry of the molecule will be used in our study.

The first representation for a protein's conformation, which we call the *representation based on angles* (**Angles**), consists of conformational states given by the three types of angles [PFK10].

The second way to represent a protein conformation, named in the following the *combined representation* (**Combined**) is based on enhancing the conformational states given by angles with the RSA values of the amino acid residues.

1.2.1.2 *Autoencoder* architecture

In the current study we use sparse denoising autoencoders to learn meaningful, lower-dimensional representations for proteins' structures, considering their conformational transitions.

We are going to use such an autoencoder in order to reduce the dimensionality of our data. Considering that one of our purposes is to be able to visualize our datasets, all the techniques implied are going to encode the protein representations into 2 dimensional vectors.

1.2.1.3 Evaluation measures

In order to determine whether the representation learned by the autoencoder preserves the evolutionary connections found in the original protein data, as well as to identify if structural similarity (given by conformational transitions) between proteins is conserved, we defined two similarity measures. Firstly, the intra-protein similarity measure, *IntraPS*, evaluates the degree of similarity between conformations within a protein and we will use this as an indication of how well the intra-protein conformational relations are maintained in the lower-dimensional representation learned by the autoencoder. Secondly, the inter-protein similarity measure, *InterPS*, evaluates the extent of correlation between two proteins, in their considered representations and it will be used to identify how well these correlations are kept in the resulting data, after applying the autoencoder. Both these measures are based on the cosine similarity measure, which is employed to evaluate the likeness between two conformations of a protein.

1.2.2 Results and discussion

The experiments we performed for highlighting the potential of deep autoencoders to capture the proteins' structure will be further presented, using the experimental methodology presented in Section 1.2.1.

1.2.2.1 Datasets

The proteins used for analysis are: 1P1L - component of sulphur-metabolizing organisms and 1JT8 - protein involved in translation [BWF+00]. These were chosen based on data availability (conformational transitions *and* RSA values), the fact that they have the same sequence length (which enables us to carry out our investigations related to RQ2) and 42 equivalent positions in common (out of the total length of 102). [YG04].

1.2.2.2 First experiment

The experiment described below is conducted with the aim of answering our first research question (RQ1) regarding the potential of *autoencoders* to unsupervisedly learn the structure of proteins as well as their ability to uncover relevant biological patterns. At the same time, we are investigating if and how the internal representation for a protein impacts the learning process.

For each protein dataset described in Section 2.2.1.1, we trained a number of denoising sparse autoencoders (Section 2.2.1.1). For the autoencoder we have employed the Keras implementation available at [C+15].

The autoencoders presented in Section 2.2.1.1 are used to reduce the dimensionality of our data and to visualize the protein datasets. Figures 1.3 and 1.4 depict the visualization of the proteins from our dataset using trained sparse denoising autoencoders.

The original data fed to the autoencoder for each protein represents a timely evolution of the protein's structure. From one conformation to another, the protein might remain unchanged, or certain parts of it might incur minor modifications. After the autoencoders have been trained, the two-dimensional representations of the proteins that they output manage to capture, as seen in Figures 1.3 and 1.4, this evolution: successive conformations

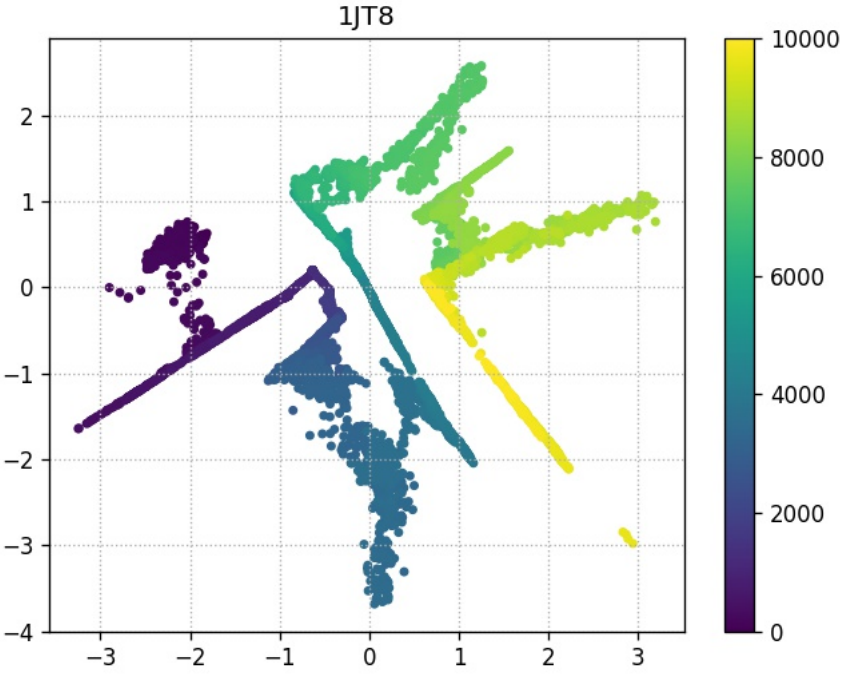


Figure 1.3: Visualization of protein 1JT8 using a trained sparse denoising autoencoder.

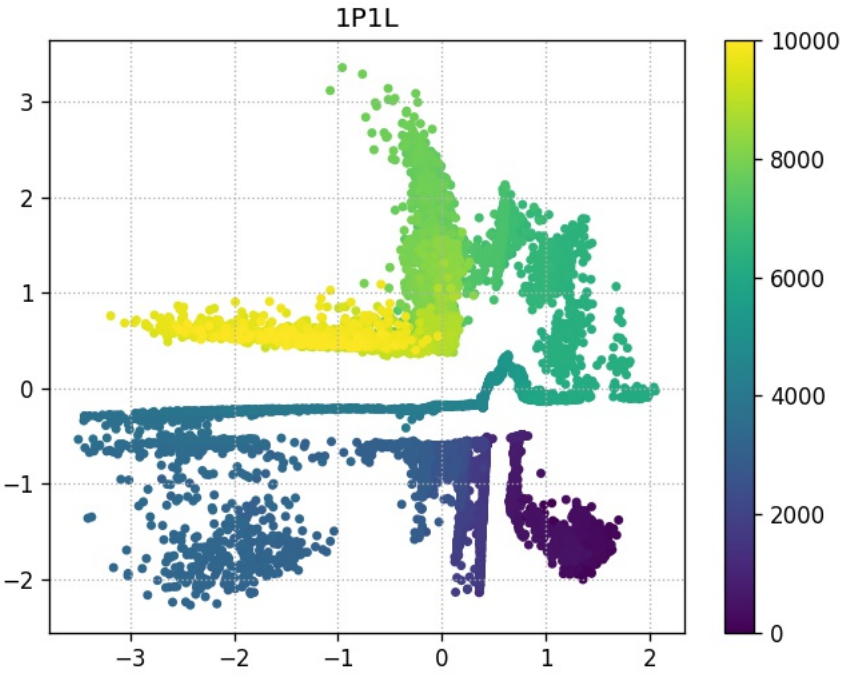


Figure 1.4: Visualization of protein 1P1L using a trained sparse denoising autoencoder.

in the original data are progressively chained together in the autoencoder's output data, thus confirming that the autoencoder accurately learns biological transitions.

Further, to decide whether the autoencoder maintains the relationships found within

the original data, we use the *IntraPS* measure. Thus, first we compute these similarities for the original data and then for the two-dimensional data output by the autoencoder, for both considered representations. The results are shown in Table 1.2. For each protein, in addition to the values for the *IntraPS* measure, we also present the *minimum* (**Min**), *maximum* (**Max**) and *standard deviation* (**Stdev**) of the cosine similarities between two consecutive conformations, for both representations.

Protein		**Angles**	**Combined**	**Min/Max/Stdev (COS)**	
				Angles	**Combined**
1JT8	Original	**0.9960**	**0.9913**	0.9894/0.9995/0.0023	0.9843/0.9962/0.0022
	Encoded	**0.9939**	**0.9985**	0.9213/0.9999/0.0161	0.9573/0.9999/**0.0044**
1P1L	Original	**0.9779**	**0.9573**	0.9593/0.9896/0.0064	0.9464/0.9695/0.0054
	Encoded	**0.9912**	**0.9962**	0.9315/0.9999/0.0119	0.9661/0.9999/**0.0052**

Table 1.2: *IntraPS* for proteins 1JT8 and 1P1L, using the two considered representations.

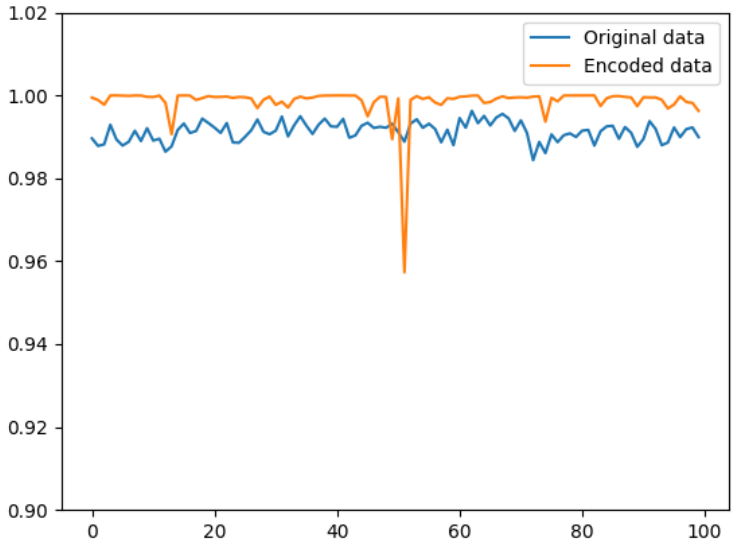


Figure 1.5: Comparative average similarities, for original and encoded data, for protein 1JT8 (combined representation).

With regard to the used internal representations, considering the results obtained for this experiment, we conclude that these do not seriously influence the learning process. This may be due to the significant reduction of data dimensionality (two dimensions). Still, for the *combined representation* which is richer in information than the *representation based on angles*, slightly better results were obtained.

1.2.2.3 Second experiment

For answering our second research question (RQ2) which was stated at the beginning of the Chapter, experiments were carried out to test if an autoencoder is capable to learn the structural similarity between proteins. In this case, we focused on the *combined representation* for the proteins, due to the fact that it seemed to lead to slightly better results (Subsection 1.2.2.2).

The results are presented in Table 1.3.

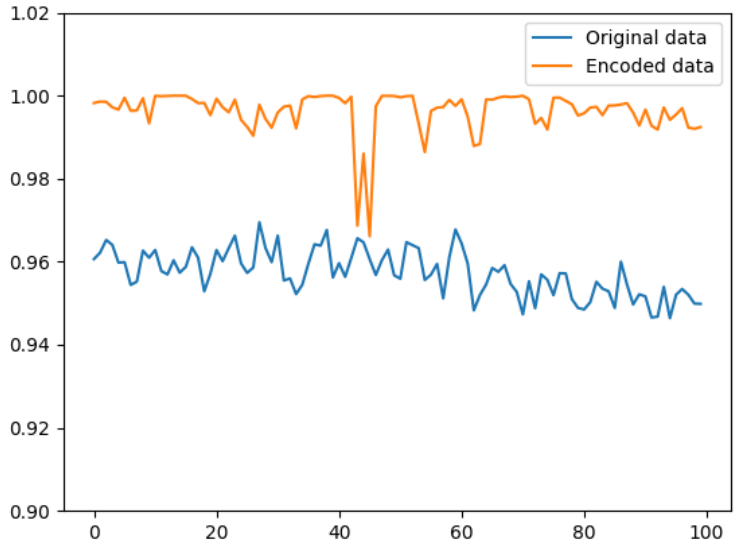


Figure 1.6: Comparative average similarities, for original and encoded data, for protein 1P1L (combined representation).

	Combined representation
$InterPS$(1JT8, 1P1L)	**0.5438**
$InterPS$(<1JT8 encoded>, <1P1L encoded>)	**0.6367**
$InterPS$(<1JT8 encoded with 1P1L>, <1P1L encoded with 1JT8>)	**0.6478**
$InterPS$(<1JT8 encoded with 1P1L>, <1P1L encoded>)	**0.6959**
$InterPS$(<1P1L encoded with 1JT8>, <1JT8 encoded>)	**0.6297**

Table 1.3: *InterPS* for proteins 1JT8 and 1P1L, using the original and encoded data.

From Table 1.3 we observe that the protein similarities computed on all combinations of the encoded data are larger than the similarities between the original data. This suggests that existing original similarities are preserved, but the dimensionality reduction induces more similarities between the encoded conformations.

1.2.3 Conclusions and further work

We have conducted a study towards applying *deep autoencoders* for a better comprehension of protein dynamics. The experiments conducted on two proteins highlighted that *autoencoders* are effective unsupervised models able to learn the structure of proteins, as well as to uncover the structural similarity between proteins. Moreover, we obtained an empirical evidence that autoencoders are able to encode hidden patterns relevant from a biological perspective.

Based on the study performed in this chapter we aim to advance our research towards predicting protein conformational transitions using supervised learning models.

Chapter 2

New Machine Learning Models for Inter-Protein Data Analysis

Proteomics is nowadays one of the most important and relevant fields from computational biology, raising a lot of challenging and provocative questions. Gaining an understanding of protein dynamic and function as well as obtaining additional insights into the protein folding process is still of great interest in bionformatics and medicine.

In this Chapter we pursue the inter proteins analysis research direction. Thus we're presenting three main original contributions that were published in original papers [TCB19, TCC19, CCT19]:

- In Section 2.1 we examine the usefulness of applying partitional and hierarchical clustering as unsupervised classification methods for uncovering proteins' structural similarity, based on the information contained within their conformational transitions. We investigate three representations for a protein based on the probability distributions of certain structural elements within conformational transitions and apply clustering methods to unsupervisedly classify proteins based on their structural similarity. Experiments are performed on two protein datasets and the obtained results are analyzed and compared with the results of similar existing approaches. The comparative results reveal that in many cases our proposal performs better than an earlier work in this topic

- In Section 2.2 we investigate the problem of supervisedly classifying proteins according to their structural similarity, based on the information enclosed within their conformational transitions. We are proposing *AutoSimP* approach consisting of an ensemble of *autoencoders* for predicting the similarity class of a certain protein, considering the similarity class predicted for its conformational transitions. Experiments performed on real protein data reveal the effectiveness of our proposal compared with similar existing approaches.

- In Section 2.3 we introduce a new approach *AnomalP* for detecting anomalous protein conformational transitions using *deep autoencoders* for encoding information about the structural similarity between proteins belonging to the same superfamily. Experiments are conducted on real protein data and the obtained results emphasize the potential of autoencoders to learn biological relevant patterns, such as proteins' structural characteristics and that they are useful for detecting conformations or proteins which are likely to be anomalous with respect to a superfamily. The study performed in this Section is aimed to provide better insights of proteins structural similarity, with the broader goal of learning to predict proteins conformational transitions.

Section 2.1 is structured as follows. We propose three different proteins representation techniques in Section 2.1.1. Having the goal to compare their effectiveness, we perform experiments using each of them and report the obtained performances in Section 2.1.2.

Section 2.2 represents an extension of the previous method and is structured as follows. The method that is based on an ensemble of deep autoencoders is presented in Section 2.2.1. Then, we perform experiments and report the obtained results in Section 2.2.2.

Section 2.3 presents an anomaly detection method based on deep autoencoders and is structured as follows. *AnomalP*, the approach for detecting anomalous conformational transitions is described in Section 2.3.1. We present the metrics used for evaluation in Section **??** then we discuss the performance and compare the approach against previous work in Section 2.3.3.

2.1 Using clustering models for uncovering proteins' structural similarity

In this section, we investigate the usefulness of applying partitional and hierarchical clustering as unsupervised classification methods for uncovering proteins' structural similarity, based on the information contained within their conformational transitions. The experiments depicted in this section were introduced in the original paper [TCB19].

The contribution of the section is summarized in the following. The main goal is to emphasize the efficiency of partitional and hierarchical clustering method for detecting, based on the conformational transitions of proteins, the structural relationships between them.

In summary, in this work we answer the following research questions:

RQ1 What is the effectiveness of using clustering methods to unsupervisedly classify proteins according to the structural relationships between them?

RQ2 How does the clustering based approach introduced in this paper compare to existing related work on detecting inter-proteins structural similarity?

2.1.1 Methodology

We introduce in the following the methodology our study is based on. Section 2.1.1.1 presents the vector models we are proposing for representing a protein using the probability distribution of the structural alphabet elements in the protein's conformational transitions.

2.1.1.1 Proteins' representations

We are considering in the following three vector representations for the proteins based on the distributions of the SA letters in their conformational transitions.

Let us consider that a protein $Prot$ of length n is visualized as a sequence of characters over the alphabet $\mathcal{A} = \{G, P, A, V, L, I, M, C, F, Y, W, H, K, R, Q, N, E, D, S, T\}$ of 20 letters representing amino acids: $Prot = a_1 a_2 \dots a_n$, where $a_i \in \mathcal{A}, \forall i \in \{1, 2, \dots, n\}$ [BPC+17]. For a protein, thousands of different conformations (represented as described above) obtained by molecular dynamics simulations are given. A conformation of the protein may be converted into its SA representation, with the structural alphabet SA being composed of the 25 letters:

$$SA = \{A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y\}$$

. Let us denote by l_i $1 \leq i \leq 25$ the letters from the alphabet SA.

Accordingly, for a protein $Prot$ a large number m of experimentally determined conformations is given. Thus, protein $Prot$ is characterized by a sequence of m conformations,

$$Seq_{Prot} = (cf_1^{Prot}, cf_2^{Prot}, \ldots, cf_m^{Prot}),$$

where

$$cf_i^{Prot} = (cf_{i1}^{Prot}\ cf_{i2}^{Prot} \ldots cf_{i,n-3}^{Prot}),\ cf_{ik}^{Prot} \in SA,\ 1 \leq i \leq m$$

.

First representation (R1)

The first representation for a protein $Prot$ is a frequency vector that we have previously introduced in [BPC+17] and is constructed as follows. For each of the 25 letters from the structural alphabet, the probability pr_{l_i} of occurrence of letter l_i in the conformational transitions of protein $Prot$ is computed. Thus, $Prot$ is visualized as a 25-dimensional vector containing the probabilities of occurrence of the SA symbols in the given protein, $Prot = (pr_{l_1}, pr_{l_2}, \ldots, pr_{l_{25}})$.

Second representation (R2)

Considering the conformations given for a protein $Prot$, a distribution vector can be computed, which stores information about the SA letters' distribution in the protein's conformations. We propose the following computation for the distribution vector. For each conformation cf_i^{Prot} of the protein $Prot$ and each of the 25 letters from the structural alphabet $lt \in SA$, we compute the probability p_{lt}^i of occurrence of letter lt in the conformational transition cf_i^{Prot}. Thus, protein $Prot$ may be visualized as a $25 \cdot m$-dimensional vector containing the probabilities of occurrences of the SA symbols in its conformational transitions, $Prot = (p_{l_1}^1, \ldots, p_{l_{25}}^1, p_{l_1}^2, \ldots, p_{l_{25}}^2, \ldots p_{l_1}^m, \ldots, p_{l_{25}}^m)$.

Third representation (R3)

The third representation we propose for a protein $Prot$ considers not only the probability of the SA letters in the conformational transitions of $Prot$, but also the frequence of all possible bigrams and trigrams composed by the letters from the structural alphabet.

Considering a certain SA letter l (e.g. letter A) there are 651 possible uni, bi and trigrams starting with letter l (e.g. A, AA, AB, ... AZ, AAA, AAB, ... AZZ). Accordingly, a number of 16275 uni, bi and trigrams may be formed considering all 25 SA letters. Let us denote by $Seq = (seq_1, \ldots, seq_{16275})$ the sequence of all possible n-grams ($1 \leq n \leq 3$).

A protein $Prot$ may then be visualized as an 16275-dimensional numerical vector $Prot = (prot_1, prot_2, \ldots, prot_{16275})$, where $prot_i$ represents the probability of appearance of n-gram seq_i in all m conformational transitions of the protein.

2.1.1.2 The clustering models

Considering the previously proposed vector models for a protein, we assume that we have a set of proteins $SP = \{Prot_1, Prot_2, \ldots, Prot_r\}$, each protein $Prot_i$ being represented as multidimensional numerical vector, as described above. With the main goal of grouping the proteins from SP such that a group contains proteins which are similar from a structural viewpoint, two clustering algorithms are applied: the *k-means* and HAC. The distance function used to measure the dissimilarity between the proteins is the *Euclidean Distance* between their corresponding high-dimensional vectors. Various distance functions were considered but overall the Euclidean Distance performed better, hence we are going to focus on this particular function.

2.1.2 Experimental results and discussion

Two protein datasets will be further used in our experiments, applying the methodology introduced in Section 2.1.1. Both datasets were used in a previous study [ATC18] regarding inter-protein similarity detection. Each protein from the datasets is characterized by a sequence of 10000 experimentally determined conformations (i.e. $m = 10000$). The proteins have different lengths, varying from 99 to 668 amino acids.

2.1.2.1 Datasets

Dataset D1 consists of *seven* proteins (codes: 1ASH, 1DLW, 1ECA, 1C52, 1CCR, 1APQ, 1COU in the Protein Data Bank [BWF+00]), taken from three different superfamilies (1.10.490.10, 1.10.760.10, 2.10.25.10).

Dataset D2 extends D1 and consists of 58 proteins belonging to nine different families.

2.1.2.2 Results

In order to answer research question RQ1 and to test the potential of *K-means* and HAC clustering algorithms to unsupervisedly classify the proteins according to their structural similarity, the clustering models introduced in Section 2.1.1.2 are applied on datasets D1 and D2 described above. Both protein representations introduced in Section 2.1.1.1 are used in the experiments. The resulting partitions are evaluated using the measures presented in Section 1.2.1.3.

The obtained results are depicted in Table 2.1, where, for each dataset, the best values obtained for *V-measure* and *silhouette coefficient* are highlighted. We observe relatively good values for both evaluation measures.

Dataset	**Representation**	**Clustering method**	***V-measure***	***Silhouette coefficient***
First	R1	*K-means*	1	0.60 ± 0.00
		HAC	1	0.60
	R2	*K-means*	1	0.51 ± 0.00
		HAC	1	0.51
	R3	*K-means*	1	0.59 ± 0.00
		HAC	1	0.59
Second	R1	*K-means*	0.68 ± 0.01	0.28 ± 0.01
		HAC	0.70	0.29
	R2	*K-means*	0.64 ± 0.02	0.21 ± 0.01
		HAC	0.67	0.22
	R3	*K-means*	0.66 ± 0.01	0.23 ± 0.01
		HAC	0.62	0.22

Table 2.1: Clustering results.

From Table 2.1 we observe that for the second dataset the clusters provided by HAC are better than those reported by the partitional *K-means* method, obtaining higher *V-measure* and *silhouette coefficients*. With regard to the first dataset, both algorithms lead to similar results for the evaluation measures.

Overall, we consider HAC as the best performing algorithm. Table 2.1 reveals that it obtained the best V-measure performance for both datasets. The representation which

provides the best evaluation scores (both V-measure and Silhouette coefficient) is the 25-dimensional one based on the distribution of the SA letters in all conformational transitions (R1). This is an interesting result, since representations R2 and R3 seemed to be more precise, encoding more information about a protein than R1. A possible explanation for this result may be the high dimensionality of the vector representations corresponding to R2 and R3 which may negatively affect the performance of the clustering algorithms.

2.1.3 Conclusions and future work

This work presents a study towards applying partitional and hierarchical clustering for unspervisedly classifying proteins according to their structural similarity. Our main goal was to find empirical evidence for two research questions related to inter-protein data analysis. First, considering three vector representations for a protein based on its conformational transitions encoded using a structural alphabet [PFK10], the usefulness of clustering was investigated. As a second focus, we compared our clustering approaches with similar related work.

Future work will target the investigation of alternative vector representations for a protein based on its conformational transitions. In addition, we plan to extend the experimental evaluation on other larger protein datasets for a better validation of our study's conclusions.

2.2 *AutoSimP*: An approach for predicting proteins' structural similarities using an ensemble of deep autoencoders

This section investigates the problem of supervisedly classifying proteins according to their structural similarity, based on the information enclosed within their conformational transitions. The experiments were introduced in the original paper [TCC19].

Our main goal is to introduce a supervised learning approach *AutoSimP* based on an ensemble of autoencoders for predicting the superfamily to which a protein belongs. The prediction is made based on the similarity between the protein's conformations and the conformations of the proteins from each superfamily (encoded into an *autoencoder*).

2.2.1 Methodology

Through the used *autoencoders* we aim to test their ability to preserve the structure of proteins and to analyze whether they are able to detect certain underlying structural relationships within the protein data. The experiments will be designed to test to what extent the encoded, lower-dimensional protein data is in accordance with biological truthfulness and to establish if autoencoders are efficient in learning structural characteristics of proteins.

AutoSimP involves four steps illustrated in Figure 2.2: *data representation and preprocessing*, *training* and *testing* (*evaluation*). The main steps of *AutoSimP* will be further detailed. The ensemble of models represents a set of fully convolutional neural networks that take as input an integer real numbered vector representing a conformation.

2.2.1.1 Data representation and preprocessing

Let us consider that $\mathcal{F} = \{F_1 \dots F_{nf}\}$ are protein superfamilies. A superfamily F_i consists of n_i proteins, i.e. $F_i = \{p_1^i \dots p_{n_i}^i\}$. For a protein p_j^i ($\forall 1 \leq j \leq n_i$) a number m of different conformations obtained by molecular dynamics simulations are given.

For each superfamily F_i ($1 \leq i \leq nf$), a dataset D_i is built using all conformations for all the proteins from that superfamily. Thus, the dataset D_i consists of $m \cdot n_i$ conformations, i.e. m conformations for each protein p_j^i ($1 \leq j \leq n_i$) from the i-th superfamily.

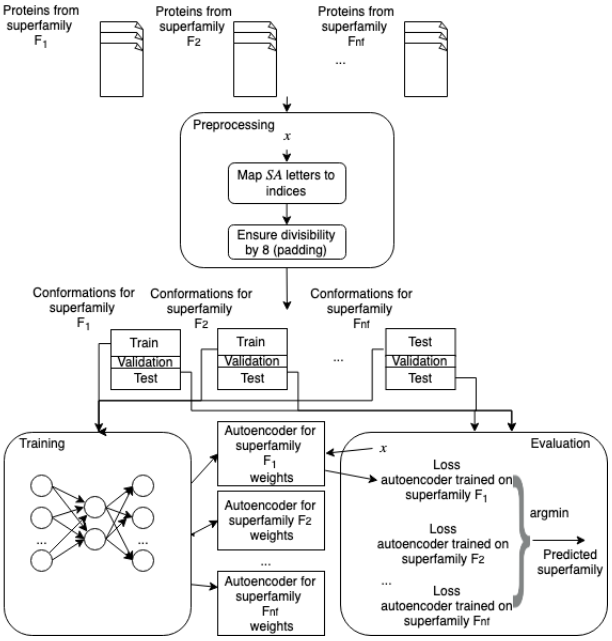


Figure 2.1: Overview of *AutoSimP* approach.

Autoencoder architecture

In the current study we use fully convolutional undercomplete autoencoders to learn meaningful, lower-dimensional representations for proteins' structures from their conformational transitions.

We are going to use such an autoencoder A_i in order to learn a lower dimensional representation for the proteins from the superfamily F_i . The main purposes is to learn meaningful representations that are specific to superfamilies.

2.2.1.2 Testing

After the ensemble of autoencoders was trained, *AutoSimP* is evaluated on 24% from each dataset D_i ($\forall 1 \leq i \leq nf$), i.e. 24% conformations for each protein from each superfamily F_j which were unseen during training.

When testing is performed at the conformation level, a conformation c is classified by *AutoSimP* as belonging to the superfamily F_i such that $i =_{j=1,nf} L_j(\widehat{c}, c)$. In the previous formula, we denoted by $L_j(\widehat{c}, c)$ the loss value computed for conformation c by the autoencoder A_j corresponding to the j-th superfamily.

When testing is performed at the protein level, a protein p represented in the testing set as a sequence $(c_1, c_2, \ldots, c_t)$ of conformations ($t = \frac{24 \cdot m}{100}$ as previously mentioned) will be classified as belonging to the superfamily F_i whose autoencoder A_i minimizes the average loss of its t conformations, i.e. $i =_{j=1,nf} \frac{\sum_{k=1}^{t} L_j(\widehat{c_k}, c_k)}{t}$.

2.2.2 Results and discussion

With the goal of answering the research questions formulated in Section , experiments on nine protein superfamilies will be conducted, using the methodology introduced in Section 2.2.1.

2.2.2.1 Results

AutoSimP approach was applied on the protein data described in Section 2.3.2.1 following the methodology introduced in Section 2.2.1. For each superfamily F_i $(1 \leq i \leq 9)$, a dataset D_i is formed by all conformations for all the proteins for the superfamily. For each protein from the dataset D_i, 6000 conformations are used for training, 1600 for validation and the remaining 2400 for testing. For highlighting the generality of *AutoSimP*, we increased its granularity level by applying it at a conformation level also, i.e. for predicting the superfamily for a certain protein conformation. Experiments were conducted on the same dataset (Section 2.3.2.1), following the same methodology as for the experiments performed at the protein level.

Level	**Measure**	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9
Protein	*Precision*	1	1	1	0.857	0.6	1	0.857	0.444	1
	Recall	1	1	0.8	1	0.5	1	0.666	0.666	1
	F-measure	1	1	0.888	0.923	0.545	1	0.75	0.533	1
Conformation	*Precision*	0.997	0.969	0.994	0.809	0.695	0.983	0.858	0.433	0.996
	Recall	0.963	1	0.803	0.955	0.508	0.999	0.638	0.696	1
	F-measure	0.980	0.984	0.888	0.876	0.587	0.991	0.732	0.534	0.998

Table 2.2: Experimental results.

The results from Table 2.2 reveal high *F-measure* values for all superfamilies, excepting F_5 and F_8. The lower performance on these superfamilies may be due to the fact that the conformations for their proteins are highly similar to conformations from proteins belonging to other superfamilies. Thus, these conformations are hardly distinguishable by the autoencoders. Further investigations will be performed in this direction.

2.2.3 Conclusions and future work

We have introduced in this paper a supervised learning approach *AutoSimP* consisting of an ensemble of *deep autoencoders* for classifying proteins in superfamilies based on the information enclosed within their conformational transitions. Experiments conducted on 57 proteins belonging to 9 superfamilies highlighted the effectiveness of autoencoders and their ability to uncover the structure of proteins and their structural similarity. The experiments have been conducted both at the protein level, as well as at the conformation level, emphasizing this way the generality of our proposal.

We plan to extend our work by applying *variational* and *contractive* autoencoders. We also aim to further study the applicability of autoencoders for detecting protein conformations that are likely to be anomalous in relation to the protein's superfamily, i.e. conformations whose structure does not resemble to their encoded information.

2.3 *AnomalP*: An approach for detecting anomalous protein conformations using deep autoencoders

In this section, we introduce a new approach *AnomalP* for detecting anomalous protein conformational transitions using *deep autoencoders* for encoding information about the structural similarity between proteins belonging to the same superfamily. The study performed in this section is aimed to provide better insights of proteins structural similarity, with the broader goal of learning to predict proteins conformational transitions. The experiments were introduced in the original paper [CCT19].

The main contribution of the study is to investigate the use of autoencoders for deciding if a certain conformation of a protein is structurally dissimilar to its superfamily, thus being

likely to represent an anomaly with respect to that superfamily. The proteins' conformational transitions are represented in our study using letters from the *structural alphabet* (SA) introduced by Pandini et al. [PFK10]. The research questions which represent the focus of our work are the following:

RQ1 To what extent may autoencoders be used for detecting protein conformations that are likely to be anomalous in relation to the protein's superfamily, i.e. conformations whose structure does not resemble to their encoded information?

RQ2 How does the representation of the proteins' conformations influence the predictive performance of the detection process?

RQ3 How to apply the approach introduced for answering the previous research questions at a protein level, i.e. to decide if a protein is likely to belong to a certain superfamily considering the dissimilarity between its conformational transitions and the encoded structural information about the superfamily?

For answering the first research question, we are introducing a supervised learning approach *AnomalP* used for providing the likelihood that a protein conformation is anomalous with regard to the protein superfamily. The prediction is made considering the dissimilarity degree of the conformation with respect to all the conformations of the proteins from the given superfamily, as encoded into an *autoencoder*.

The second research question will be investigated by considering an alternative vectorial representation for the conformations by replacing the SA letters with the angles between the underlying consecutive amino acids from the protein's primary structure (i.e. the torsion angle of all four atoms [PFK10] together with the alpha carbon atoms of the amino acids).

For highlighting the generality of *AnomalP* and for answering the third research question, *AnomalP* will be applied at a higher granularity level, to decide if a certain protein belongs or not to a given superfamily.

2.3.1 Methodology

We are further introducing *AnomalP* approach for detecting anomalous protein conformational transitions using deep autoencoders for encoding information about the structural similarity between the proteins belonging to the same superfamily.

2.3.1.1 Theoretical model

The problem we are focusing one is a binary classification problem. Given a set $\mathcal{F}$ of proteins superfamilies, $\mathcal{F} = \{F_1 \dots F_{nf}\}$, we aim to predict if a certain protein conformation belongs or not to a certain superfamily F_i, namely to detect the likelihood of being anomalous with respect to F_i. For deciding if a conformation is likely to represent an anomaly with respect to F_i, we are computing the dissimilarity degree of the given conformation with respect to all the conformations of the proteins from F_i, as encoded into an *autoencoder*.

Besides a conformational level granularity, *AnomalP* may also be used at a protein level for predicting if a protein (characterized by its conformational tansitions) belongs or not to a given superfamily.

AnomalP is aimed to empirically demonstrate that autoencoders are able to self-supervisedly learn the structural similarity and relationships between proteins, as well as to be used as an anomaly detector, i.e to determine if a conformation/protein is likely to be abnormal with respect to a certain superfamily. There are three main stages of *AutoSimP* as illustrated in Figure 2.2:

1. **Data representation and preprocessing**.

2. **Training**.

3. **Performance evaluation (testing)**.

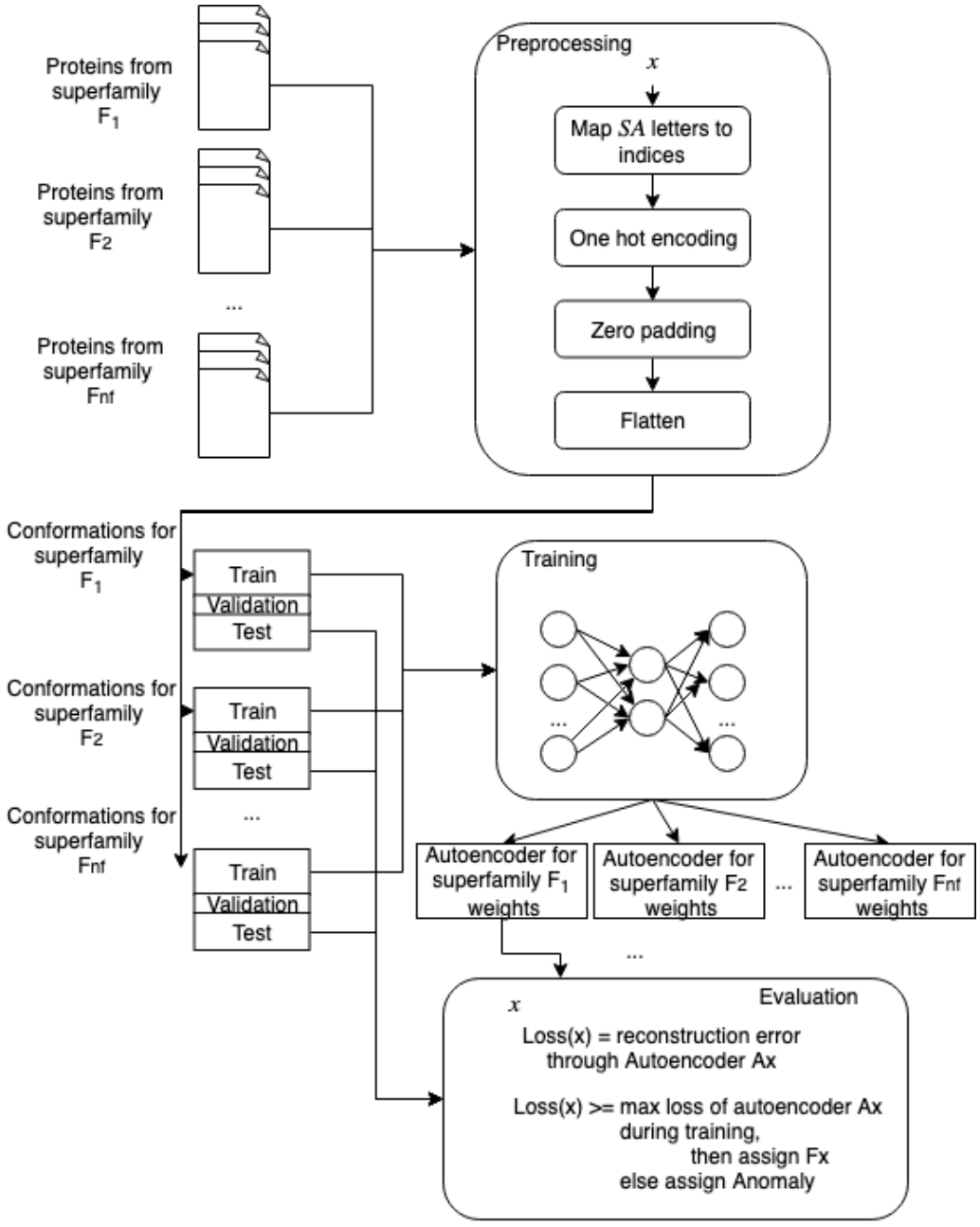


Figure 2.2: Overview of *AnomalP* approach.

Autoencoders architecture

The autoencoders employed for learning the patterns for each specific superfamily of proteins are formed by an encoder and a decoder both fully connected. The hidden layer is of size 10 in the case when rank-based representation is used, and respectively 25 when the angles-based representation is used. We experimented with other intermediate sizes too and tried to reduce the dimensionality up until the point in which the performance was not satisfactory.

The loss function which is used for training the autoencoders is the *binary cross entropy*. For the rank-based representation, this function is also used when computing the probability of anomaly. However, for the angles-based representation, the mean absolute error is monitored together with the binary cross entropy during training and the former is used when computing the probabilities.

2.3.1.2 Classification using *AnomalP*

After *AnomalP* classification model has been built, at the testing stage, we decide if a certain conformation c is likely to be anomalous with respect to superfamily F_i if the loss of autoencoder A_i computed for that conformation is greater than the maximum loss obtained during the training of A_i. Intuitively, this means that the conformation c is likely to be dissimilar to the structural information encoded in A_i and characterizing superfamily F_i.

2.3.2 Experimental evaluation

With the goal of answering research questions RQ1 and RQ2 formulated in Section , an experimental evaluation of our *AnomalP* classifier for detecting anomalous conformational transitions is further provided. The performance of *AnomalP* is experimented on nine protein superfamilies, following the testing methodology introduced in Section 3.1.1.2.

2.3.2.1 Dataset

The dataset used in our experiments consists of 57 proteins belonging to 9 superfamilies and was previously used in the literature for intra- and inter-protein analysis [ATC18, TCB19]. The dataset was obtained from the MoDEL database available at [MDH+10].

2.3.2.2 Results

AnomalP models were applied on the protein data described in Section 2.3.2.1 following the methodology introduced in Section 2.3.1.

2.3.3 Discussion

An analysis of the experimental results from Section 2.3.2.2 obtained by applying *AnomalP* classifier for detecting protein conformational transitions which are likely to be anomalous is provided in Section 2.3.3.1.

2.3.3.1 Analysis of *AnomalP*

The experimental evaluation of our *AnomalP* proposal performed on real proteins (Section 2.3.2.2) highlighted its very good performance in detecting the likelihood of a conformational transition of a protein to be an anomaly with respect to the protein's superfamily. The assumption that we used in our current evaluation that the superfamily of the protein is known when applying *AnomalP* is not a limitation of our proposal. In a real scenario, in which we are simply given a conformational transition of a protein, without knowing its superfamily, we may apply beforehand an approach for determining the protein's superfamily. We have previously introduced such an approach, call *AutoSimP* [TCC19], which uses an ensemble of autoencoders for encoding the structural information about proteins' superfamilies with the aim of predicting the superfamily of a new protein.

From Figure 2.3 and Table 2.2 we observe a weaker performance of *AnomalP* using the *angles-based* representation, both in terms of *precision* and *AUC*, on superfamilies F_2 and F_8. The lower performance on these superfamilies may be due to the fact that the information encoded about the conformations for their proteins are highly similar to

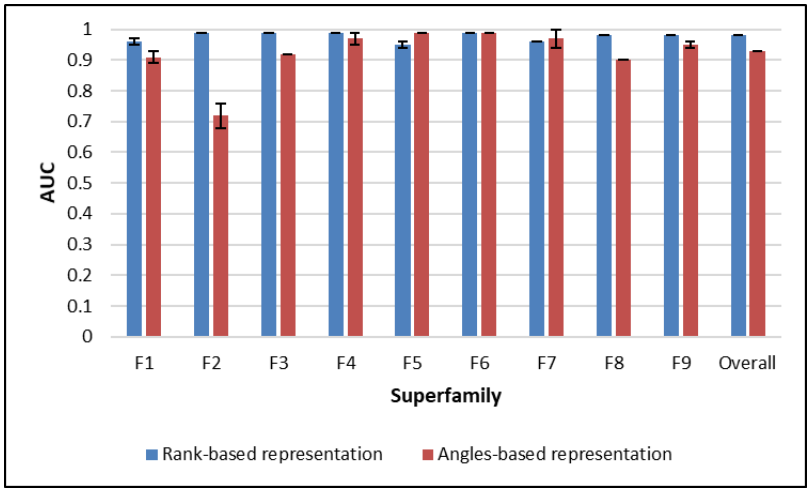


Figure 2.3: AUC values obtained by *AnomalP* using the *rank-based* and the *angles-based* representation for the conformations. The error bars represent 95% CIs.

conformations from proteins belonging to other superfamilies. Thus, these conformations are hardly distinguishable by the autoencoders. Further investigations will be performed in this direction.

2.3.4 Conclusions and future work

In this section we proposed a self-supervised learning based approach *AnomalP* which used autoencoders for determining if a certain protein conformation is structurally dissimilar to the protein's superfamily, thus being likely to represent an anomaly. The generality of *AnomalP* has been highlighted by its applicability at the protein level, not only at the conformation level. The experimental evaluation performed on real proteins belonging to nine superfamilies revealed a very good predictive performance for the proposed approach. Thus, we obtained empirical evidence that autoencoders are able to accurately encode relationships between proteins included in the same superfamily.

Chapter 3

Deep Learning Models in Computer Vision

Image classification is a well known example of a complex problem that can be tackled using deep learning. In the latest period, convolutional neural networks were successfully applied in order to solve the problem. Thus, recent development of deep learning approaches have improved the performance of the visual recognition systems.

From a computer vision perspective, developing an image classifier consists of writing an algorithm that is able to classify images into distinct classes. For improving the robustness of these classifiers, researchers have proposed a data-driven approach. So, from a Machine learning perspective, instead of explicitly describing what every image category looks like, we provide the algorithm with labelled samples for each class of image class. The learning algorithm is going to fine tune its parameters in order to learn the visual appearance of each class of image.

In this Chapter we're highlighting three main original contributions that were published in original papers [Tel17, TD19, DT19, TC20]:

- In Section 3.1 we investigate the possibility to use some supervised learning techniques in order to build models capable to accurately perform handwritten signature analysis. The results reported during the testing phase of the obtained model are encouraging for further work. Deciding whether a handwritten signature is legit or it has been falsified is a very complex task. Several methods have been tried out by the graphology experts in order to detect such fraud. However, it is obvious that it is very hard to perform such a classification.

- In Section 3.2 we propose a lightweight solution for solving an image classification problem, namely fruit recognition. The solution is tested on an open dataset. We achieve state of the art performance in terms of both classification accuracy and execution speed. We observe that recently, the research directions have been focusing more on developing lightweight models that can still achieve good classification performance.

- Besides image classification, there is another important research direction, namely object localization. Thus, in Section 3.3 we present a novel deep learning based preprocessing method to jointly detect and deskew documents in digital images. Our work intends to improve the optical recognition performance, especially on frames which are skewed (slightly rotated) or have cluttered backgrounds. The proposed method achieves good document detection and deskewing results on a dataset of photos of cash receipts.

- We introduce in Section 3.4 a new approach for grouping proteins based on their internal structure similarity. We propose a system based on *one-shot learning* and *siamese convolutional neural networks* for dealing with this task. The system analyses graphical representations of proteins obtained from the Protein data bank for clustering them together based on their structural similarities. The experimental results highlight that $CVSimP$ outperforms, in terms of $F\text{-}measure$, similar related work from the literature.

Section 3.1 is structured as follows. We firstly discuss the difficulty and importance of this problem. Then we present the method in Section 3.1.1 and the obtained results in Section 3.1.2.

Section 3.2 is structured as follows. We discuss a solution for an open source dataset for fruits recognition is presented in Section 3.2.1. The method is based on lightweight neural models and, as shown in Section 3.2.2, achieves state of the art performance for both execution time and classification performance.

Section 3.3 presents a *convolutional neural network* based localization technique and is structured as follows. We present the approach in Section 3.3.1 and then we depict and compare the obtained results in Section 3.3.2.

Section 3.4 is structured as follow. In Section 3.4.1 we describe our approach. We're going to cover aspects regarding the topology of the model, the loss function used and evaluation. After that, in Section 3.4.2 we're going to describe the dataset used and the parameter settings of the experiment. We're also showing some results and do a discussion related to a previous study. Finally, the conclusions and future improvement ideas are discussed in Section 3.4.3.

3.1 Detecting false signatures using Convolutional Neural Networks

In the current section, we describe an experimental pipeline for analyzing the authenticity of handwritten signatures. The methods and experiments depicted in this section were introduced in the original paper [Tel17].

The aim of this section is to present a *machine learning* approach proposed for this binary classification, to highlight its performance by testing it against new, unseen data.

3.1.1 The proposed approach

We consider that this problem is solvable in a supervised manner since we can use already annotated datasets of images of signatures. In this learning scenario, the model will learn to detect whether an image contains a legit signature or a false one, by analyzing such already annotated examples.

Our approach consists of three steps. First, a *feature extraction* step is applied on the input data. For this step we are using the *Tensor flow Inception graph* pretrained model [SVI+15]. This model was developed by *Google* and it represents a very complex convolutional neural network which is composed of 59 layers. The model was trained on a considerable set of images and was capable to obtain state of the art accuracies on very complex problems, such as *ImageNet* classification [SVI+15].

The next step consists of training a classifier on the pre-processed data. More specifically, using the features extracted from our dataset of signatures we aim to build a classifier that will learn to identify the forger signatures based on such input data. A *Support Vector Machine* classifier will be used for discriminating between original and false signatures. The trained *SVM* will be then *tested* in order to evaluate its performance.

3.1.1.1 Training

On the set of extracted features, a *SVM* is trained. In order to do so, we take all the available samples in the dataset and we apply the feature extractor. Furthermore, the obtained set of instances (vectors of features) is split in 2 sets: training and testing.

In order to train the model, several hyperparameters are used, such as C, the kernel function, the parameters of the kernel (e.g. γ for the RBF kernel). For optimizing the hyperparameters, a grid search is performed in order to find the best suited ones on a 10-fold cross validation approach.

3.1.1.2 Testing

The performance of the trained *SVM* model will be tested on a testing set completely disjoint from the training dataset. The testing phase will be performed on unseen data.

Since the considered problem is a binary classification one, the *confusion matrix* will be computed. For building the confusion matrix and computing the measures, we consider that the *forger* signatures are representing the *positive* class while the *negative* class is represented by the *original* ones. A large number of different performance metrics can be computed from the confusion matrix.

We report both the *accuracy* and the *confusion matrix* related measures because by doing so we can easier interpret the performance of the model. Moreover, since the testing set is imbalanced, these measures can be considered very important.

3.1.2 Results and discussion

3.1.2.1 Dataset and parameters setting

The dataset used in our experiments is free and publicly available [KC13]. It consists of 4000 annotated samples from which 800 are forgeries. In order to construct the dataset, several persons were asked to write down their own signature. Furthermore, another person was asked to try to replicate the original signature.

In the dataset we have multiple signers each of them having the original signature and some forgeries. We intend to train our model in order to distinguish between the two signature types, forgery and original in an offline manner [KC10b].

3.1.2.2 Results

For our experiments we have used the *scikit-learn* implementation of *SVM* [PVG+11]. 80% of the dataset was reserved for training and on these instances we performed a training methodology which mainly consisted of a *SVM* grid search over the training set. The testing methodology described in Section 3.1.1.2 was applied on the trained *SVM* using the rest of the dataset. The obtained *accuracy* (*Acc*) was **95.1%**.

For our experiment, the reported 95% CI for the *accuracy* on the testing set is **[0.935, 0.966]**. Thus, there is a 95% confidence that the accuracy of our classifier ranges in the confidence interval.

The AUC measure computed for our classifier is **0.92** and the *F-measure* is **0.88**. These values express a very good performance for the proposed classification model.

The dataset was reshuffled in order to repeat the random split for the training and testing sets. The proposed experiment was repeated 20 times in order to analyze the evolution of the AUC measure.

If we look only to the performance measure of the approaches described in Table 3.1, we observe that our approach is comparable to the related work. Moreover, the 95% CI obtained by our approach is very small, compared to the one from [RGSK11], and this proves again the performance of our model.

#	Approach	Performance
1	**Our approach**	$95\% \pm 0.015$
2	Statistical analysis [KC10b, KC10a]	89%
3	Statistical learning[SSB06]	84%
4	Deep learning[RGSK11]	$85.03\% \pm 14.25$

Table 3.1: Comparison to related work based on the accuracy evaluation measure.

3.1.3 Conclusions and further work

In this paper we have presented a *machine learning* method based on a feature extractor that can be successfully used in solving the signature verification problem. Considering the good results, we may say that we have confirmed again that this complex task is suitable for *machine learning* solving.

Further work consists in extending the experiment on multiple benchmark datasets in order to have a better overview of the capability of the proposed method. Building a convolutional neural network from scratch will be also considered.

3.2 Lightweight deep learning models for fruits recognition

The purpose of this section is to give an efficient and accurate solution for an image classification problem. More exactly, we are going to focus on a specific problem, namely fruit recognition. The methods and experiments depicted in this section were introduced in the original paper [TD19].

We are going to study the fruit recognition problem described in [MO18] with the general purpose of developing a state of the art model. On the other hand, we want to tackle the speed-accuracy trade off so we are going to focus exclusively on lightweight models. Finally, we intend to conduct a discussion regarding the dataset and give some possible future direction for the discussed problem.

The following research questions will be investigated in this section:

RQ1 How to design a lightweight deep learning model for improving the state of the art performance in fruits recognition from images?

RQ2 How does the proposed learning model compare in terms of performance and execution time to the existing state of the art models?

3.2.1 Our approach

Our solution to the fruits recognition problem is discussed in this section, with the goal of answering research question RQ1.

3.2.1.1 Dataset

For our experiments, we use the dataset *Fruit-360* [1] described in [MO18]. The dataset consists of thousands of images of fruits. The authors propose 81 classes of fruits (e.g. apple red, orange, guava, plum, raspberry etc).

[1] accessed November, 2018

3.2.1.2 Machine learning methodology

Let us consider $I = \{i_1, i_2, \ldots, i_n\}$ a set of images representing fruits and $C = \{c_1, \ldots, c_k\}$ a set of classes of fruits (e.g. apricot, banana). Thus, our classification problem is formalized as approximating, from the training dataset I, a target function $f : I \to C$ that maps instances from I to classes from C. The learned approximation $h \approx f$ is called *hypothesis*.

For solving the problem, we are designing variants of convolutional neural networks. These neural networks are capable to take an image of a fruit as input, extract features and predict its type.

Our models are composed of two main parts: a feature extractor (the backbone of the network) and the output layer. For building the feature extractor, we adapt state of the art models in terms of speed-accuracy trade off: MobileNet V2 [SHZ+18] and ShuffleNet V2 [MZZS18].

After the feature extractor a global average pooling layers is applied [LCY13]. Finally, an output layer composed of a vector of 81 logits normalized by a softmax function [GBC16] is used.

Another common approach that we find interesting and usually very useful is *transfer learning*. In a transfer learning methodology [GBC16] one can make use of a pretrained model in order to build a model that is fit to the new dataset.

3.2.1.3 Dataset augmentation

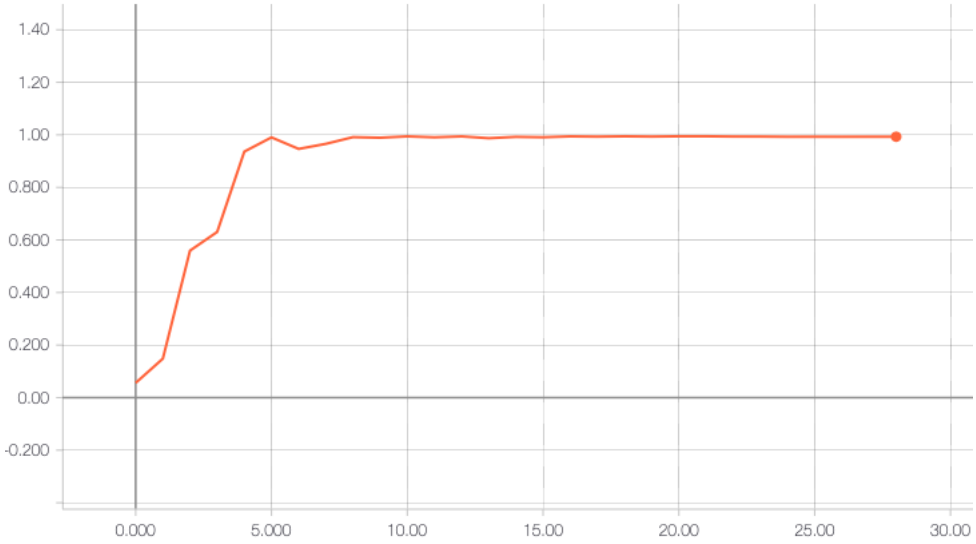


Figure 3.1: Evolution of validation accuracy for a particular experiment run on MobileNet V2 that is trained from scratch.

In order to improve the performance of the model we make use of some data augmentation techniques. For our problem setting we use only horizontal and vertical flips since we observed that all the samples in the dataset are centered.

3.2.2 Experimental evaluation

In this section, the results achieved by our models are presented. Moreover, the purpose of the section is to compare our method to related work and to show its advantages in terms of speed and accuracy.

Model backbone	Transfer learning	Augmentations	Test accuracy	Best Test accuracy	Worst Test accuracy
MobileNet V2	No	No	$97.3\% \pm 0.3$	98.0%	96.4%
MobileNet V2	No	Yes	$98.0\% \pm 0.2$	98.5%	97.4%
MobileNet V2	ImageNet	No	$98.6\% \pm 0.2$	98.9%	97.9%
MobileNet V2	ImageNet	Yes	$98.7\% \pm 0.1$	99.1%	98.2%
ShuffleNet V2	No	No	$97.6\% \pm 0.3$	98.2%	96.9%
ShuffleNet V2	No	Yes	$98.4\% \pm 0.1$	98.8%	98.1%

Table 3.2: Our test results for various settings. 95% CIs are used.

3.2.2.1 Results and analysis

We perform several types of experiments for finding the best performing model. Different experiment settings are combined in order to find the most suitable methodology. For instance, we want to highlight the impact of geometric augmentations (i.e. random horizontal and vertical flips).

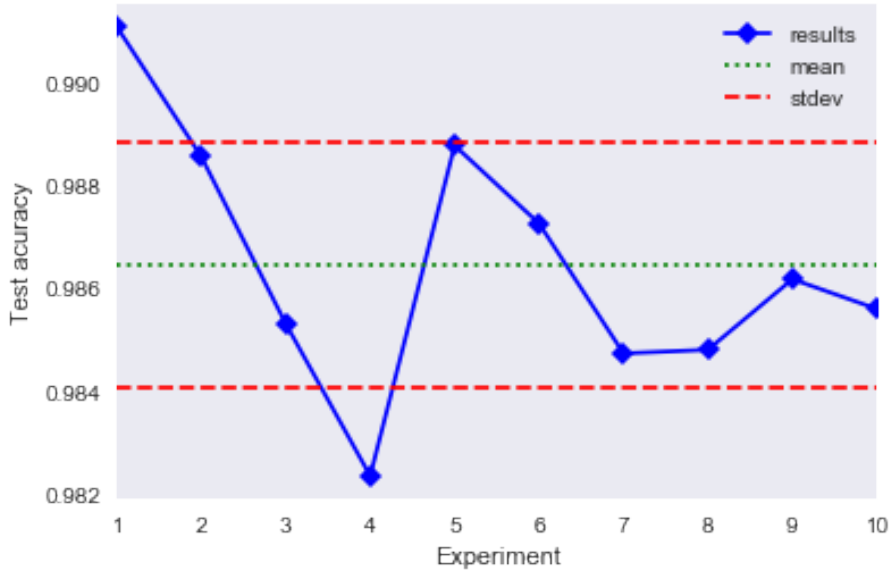


Figure 3.2: Test accuracies of the 10 experiments obtained by MobileNet V2.

In Figure 3.1 we present the evolution of the validation accuracy for a particular test run on the MobileNet V2 that is trained from scratch. No augmentations are used for this experiment. We observe that the model manages to achieve a good validation accuracy relatively fast.

3.2.2.2 Comparison to related work

For answering research question RQ2 we compare the learning model proposed in Section 3.2.1 to the existing state of the art models in fruits recognition from images. The comparison is conducted in terms of model accuracy and execution time.

The results presented in Table 3.3 show that our model outperform any other existing work. It is worth mentioning that the other authors used a former version of the dataset that had less classes. For instance, the results included in [AJW18] are reported on a version with 74 classes. As mentioned in [MO18], the dataset is continuously updated.

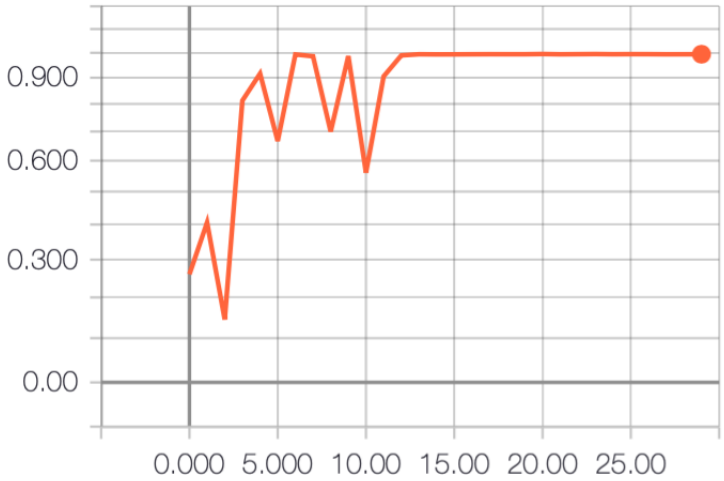


Figure 3.3: Evolution of validation accuracy for a particular experiment run on MobileNet V2 that is trained using ImageNet initialization and data augmentation.

Model	Test accuracy
Mureşan and Oltean [MO18]	96.3%
Baryła [Bar18]	90.1%
Andersson et al. [AJW18]	96.7%
Ours	$98.7\% \pm 0.1$

Table 3.3: Comparison to related work in terms of test accuracy

3.2.3 Conclusions and future work

In this study we studied the fruit classification problem using a benchmark dataset. We have successfully conducted our study towards proposing lightweight models for this problem. Moreover, as shown in Section 3.2.2 the employed models achieved state of the art performance regarding both classification metrics and speed. The most important success of our study is that one can easily use our models for building a fast and accurate mobile application.

Furthermore, during the review conducted in Section 3.2.1.1 we made some observations that can be useful for extending and improving the dataset. We let this as future work. Even though the test performance obtained by our model is good enough for state of the art, it's worth mentioning that our methodology can be extended. One could try other experiment settings such as different depth multiplier values. On the other hand, it would be worth considering a pretrained version of ShuffleNet V2 when initializing that particular network.

3.3 A document detection technique using convolutional neural networks for optical character recognition systems

In the current section, we introduce a novel document preprocessing method for optical character recognition. The methods and experiments depicted in this section were introduced in the original paper [DT19].

We propose a novel preprocessing method based on document detection which uses Deep learning and projective transformation. The method is using a *convolutional neural network* to detect the key points of the document, then it makes use of these points for projecting

the document into a rectangular shape. Moreover, we show that our method is capable to both detect and skew correct images of documents.

The capacity of *convolutional neural networks* to predict bounding boxes of objects, by framing the object detection task as a regression problem has already been demonstrated with Overfeat [SEZ+13]. Our technique makes use of this capacity of *convolutional neural network*s to predict (x, y, *width*, *height*) values and adapt it for predicting multiple 2D points in the image space.

3.3.1 The proposed approach

We introduce in this section our methodology for modelling and solving the problem of document skew detection and correction in a supervised learning manner.

3.3.1.1 The proposed Machine learning model

Since we are attempting to solve an image processing problem using machine learning, we choose to use a *convolutional neural network*. We are going to build our model based on a *MobileNet* [HZC+17] backbone.

The input of the model is represented by a photography of a cash receipt. The image is encoded as a *RGB* tensor denoted as x. Given x, the model computes the location of 4 points in the image domain representing the corners of the cash receipt. Each point is represented by a pair denoted as $(\widehat{y}_1, \widehat{y}_2)$ making our model return 8 values: $< \widehat{y}_{1,1},\ \widehat{y}_{1,2},\ \widehat{y}_{2,1},\ \widehat{y}_{2,2},\ \widehat{y}_{3,1},\ \widehat{y}_{3,2},\ \widehat{y}_{4,1},\ \widehat{y}_{4,2} >$.

3.3.2 Experimental evaluation

In this section we present the experiments conducted in order to assess the effectiveness of the proposed approach. It is composed of two main parts: *training* and *testing*.

3.3.2.1 Dataset

The dataset used in our experiment consists of a collection of photos of various types of cash receipts collected from different sources. All the images are of the same high quality resolution (1920x1080). In order to decrease the complexity of the model for time efficiency reasons and without losing performance, all the images (and the corresponding labels) are down scaled to 480x270.

The input values are rescaled to $[-1, 1]$ [HZC+17]. For our experiment, we use a dataset composed of 6000 entries.

For constructing the ground truth values we have developed a simple web application which was used to manually annotate the dataset. The application presents samples of photographies to the user which is asked to mark the 4 key points of the document. After marking the coordinates, the user is capable to visualize the transformed image obtained by applying the *projective transformation* into a rectangular shape using the key points they placed.

3.3.2.2 Experimental methodology and results

The *training* is performed on 90% of the dataset. We employ a distinct validation set formed of the remaining 10%.

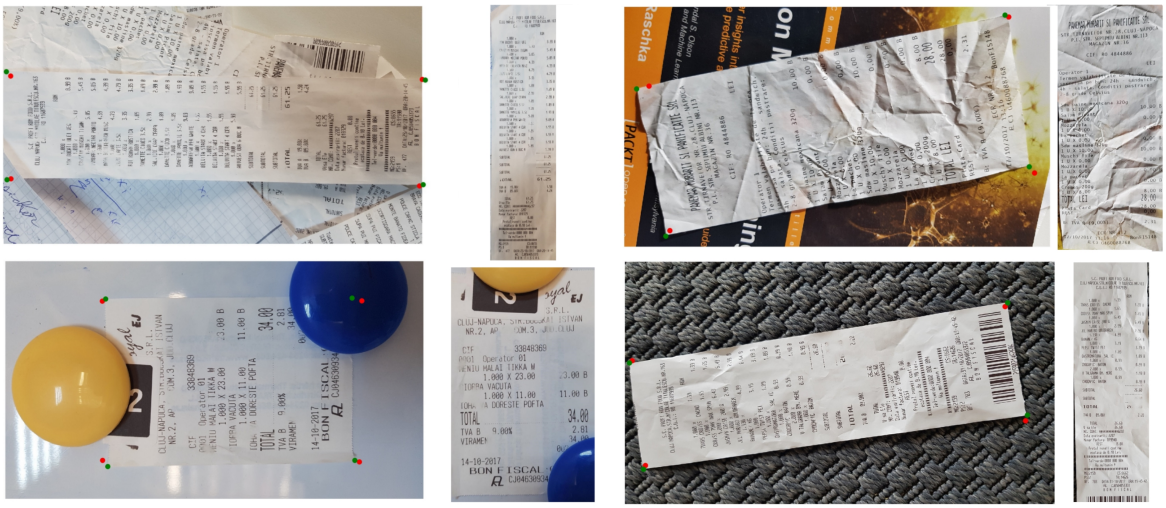


Figure 3.4: Test samples including the original image and the projection of the detected receipt. The prediction is marked by red dots and the ground truth is marked by green ones. It is worth mentioning that the model correctly marked the cash receipt even though the corner was obstructed.

The obtained model is tested against a new collection of images, the testing set. It consists of 700 images of cash receipts. The cash receipts come from different providers than those found in the training set and were designed to be very difficult. For each instance we detect the 4 key points and we report the *mean absolute error*, the *angular error* and the *absolute value* of the skew angle on the resulted projection. We perform three experiments: one version is using the classical MSE as loss while the other two versions are using the proposed loss function. The obtained results are depicted in Table 3.4. We report the MAE, the Angular error and the mean of the absolute values of the skew angles. The experiments were repeated 10 times for each $\lambda \in \{0, 1, 5\}$ in order to compute the 95% confidence intervals (CI). Test samples including the detection and the projection are depicted in Figure 3.4. The reported metrics show that the best model in terms of skew correction was obtained using the model that used the proposed loss function with $\lambda = 5$.

Model	MAE	Angular error	Hough
MobileNet $\lambda = 0$	3.66 ± 0.07	4.87 ± 0.07	1.04 ± 0.01
MobileNet $\lambda = 1$	3.37 ± 0.04	4.05 ± 0.12	0.96 ± 0.01
MobileNet $\lambda = 5$	3.38 ± 0.05	3.22 ± 0.14	0.88 ± 0.01

Table 3.4: Results with 95% CIs.

Table 3.5 depicts the results obtained by comparing our method against related work. For an accurate comparison we run all algorithms on the same test set, computing the skew angle using the Hough transform over the obtained projections. The results show an overall better performance for our method compared to the approaches from [Xio16] and [JS17].

The average inference time per frame for our model is 109ms on a OnePlus5 mobile device, using the TFLite runtime (from Tensorflow 1.12) with 8 threads.

Model	Min	Max	Mean	Std
MobileNet $\lambda = 0$	0.00 ± 0	9.61 ± 0.08	1.04 ± 0.01	1.16 ± 0.02
MobileNet $\lambda = 1$	0.00 ± 0	9.33 ± 0.07	0.96 ± 0.01	1.10 ± 0.02
MobileNet $\lambda = 5$	0.00 ± 0	9.62 ± 0.05	0.88 ± 0.01	2.20 ± 0.02
Xiong [Xio16]	0.00	9.45	1.50	1.84
Javed and Shafait [JS17]	0.00	9.79	3.75	2.80

Table 3.5: Comparison to related work based on Hough values. 95% CIs are provided for our models.

3.3.3 Conclusions and further work

We presented a new preprocessing technique effective for making images more accessible to *OCR* algorithms. The main benefit of our technique is the adaptability to unseen situations. The process combines two very important steps for obtaining good results: document detection and deskewing.

Further work will be performed on extending our method to be used for different kinds of documents. We plan to generalize the angular loss for n key points, suiting detection problems where the objects of interest appear in higher order polygonal forms.

3.4 *CVSimP*: An approach for predicting proteins' structural similarity using one-shot learning

In this study the problem of classifying proteins according to their similarity is approached, from a computer vision perspective. We introduce a new approach *CVSimP* for predicting proteins' structural similarity using *one-shot learning*. The methods and experiments depicted in this section were introduced in the original paper [TC20].

In the *deep learning* literature, *one-shot learning* is a methodology to solve object categorization problems. Most of the applications are related to computer vision. While other machine learning based classification methods require a tedious training on thousands of samples thus very large datasets, the models based on one-shot learning can be trained using less training samples. One-shot learning models are developed using *siamese neural networks* that can predict if two instances are belonging to the same category or not.

Siamese neural networks were firstly proposed by Bromley et al. [BGL+94]. The authors suggested this novel architecture of neural networks for discriminating pictures of signatures based on their author. A siamese neural network is composed by two networks which accept two different inputs, those networks being joined at the top.

To sum up, in this work we are trying to answer the following research questions:

RQ1 How to introduce an *one-shot learning* approach for predicting the structural similarity of proteins from protein images? In this respect, a new approach *CVSimP* will be introduced and empirically validated.

RQ2 How does the *one-shot learning* approach proposed in this work compare to existing similar work on detecting inter-proteins structural similarity?

3.4.1 Methodology

With the goal of answering our first research question RQ1, we introduce in this section the methodology that we use for solving the classification problem. We are going to define the problem and describe our machine learning model. We're also covering aspects regarding training and performance evaluation.

We are going to make use of raw protein representations. We consider that these raw representation may leverage better performance for our convolutional neural networks based classifier.

The problem may be viewed as a binary classification task. We will consider pairs of protein images as input with the goal of predicting if these represent the same protein or not.

Model

Our model is a siamese convolutional neural networks. The architecture of the model is depicted below:

$$
\begin{aligned}
&INPUT(100x100x3) \rightarrow \\
&\rightarrow CONV(3x3, 6) \rightarrow ReLU \rightarrow BatchNORM \rightarrow \\
&\rightarrow [CONV(3x3, 8) \rightarrow ReLU \rightarrow BatchNORM] \cdot 2 \rightarrow \\
&\rightarrow FC(500) \rightarrow ReLU \rightarrow FC(250) \rightarrow ReLU \rightarrow OUT(8)
\end{aligned}
$$

Loss function

For training the model, having the primary goal to optimize the encoded output vectors, we employ a *contrastive loss* [HCL06]. The loss function was proposed by Hadsell et al. and represents a method to teach the model to learn mappings in such a way that the samples marked as similar are pulled together while the ones marked non-similar are pushed apart. The contrastive loss makes use of the euclidean distance that can be defined as the L_2 norm of the difference between the two vectors.

3.4.2 Experimental results and discussion

In this section we are describing our case study on a dataset of images obtained from Protein data bank [BWF+00, BHN04], aiming to further answer research questions RQ1 and RQ2. The Machine Learning pipeline and the obtained results are further described. First, the dataset used in our experiments is described in Section 3.4.2. Then, we continue with presenting in Section 3.4.2 the performed experiments and in Section 3.4.2 the obtained results.

Dataset

Our dataset consists of images obtained from 57 proteins. For each protein, we have captured various views from different angles. All samples have black background and the protein is centered. These proteins are classified as belonging to 9 superfamilies as revealed in Table 3.6. For each superfamily, the third column depicts the proteins which are used for training the model and the last column represents the protein used for testing.

We have collected image data for each protein introduced in our study from the *Protein data bank* [BWF+00, BHN04]. The representations are created using *NGL* [RBV+18].

Experiments

The model is trained for multiple epochs on pairs created using graphical representation of proteins from the training set and it is evaluated at the end of the each epoch using a fixed validation set. The pairs used for training are recreated randomly after the end of each epoch. For developing the model and the training and evaluation method we have used *PyTorch* [PGCC17].

#	**Superfamily**	**Proteins** used for training	Testing protein
1	**3.20.20.80**	{1B1Y, 1CNV, 1ITX, 1JFX, 1KFW, 1NAR, 1VFF, 2EBN}	$1EDG$
2	**1.10.490.10**	{1ITH, 1MBA, 2HBG, 2LHB, 1ASH, 1DLW, 1ECA}	$1HLB$
3	**1.10.238.10**	{1OMR, 1SRA, 2SAS, 1CB1, 1IQ3}	$1UHN$
4	**2.40.50.140**	{1SLJ, 1YVC, 1EOV, 1JT8, 1KRS}	$1AH9$
5	**2.60.120.260**	{1NKG, 1PMJ, 1GUI, 1I5P, 1K45}	$1ULO$
6	**3.30.30.10**	{1PE4, 1SEG, 1BCG, 1GPT, 1I2U}	$1JXC$
7	**2.60.40.10**	{1R6V, 2FCB, 1JBJ, 1JE6, 1NCT}	$1OLL$
8	**3.40.50.150**	{1Y8C, 1DUS, 1F3L, 1YUB}	$1AF7$
9	**2.160.20.10**	{1RU4, 1VBL, 1BHE, 1EE6}	$1QCX$

Table 3.6: Proteins considered for this study [ATC18].

Approach	**Evaluation measure**	$1EDG$ (F_1)	$1HLB$ (F_2)	$1UHN$ (F_3)	$1AH9$ (F_4)	$1ULO$ (F_5)	$1JXC$ (F_6)	$1OLL$ (F_7)	$1AF7$ (F_8)	$1QCX$ (F_9)	**Overall**
$CVSimP$	*F-measure*	0.967	0.835	0.828	0.905	0.848	0.901	0.856	0.828	0.858	0.867
Autoencoders [TCC19]	*F-measure*	1	1	0.888	0.923	0.545	1	0.750	0.533	1	0.845
Clustering [TCB19]	*F-measure*	-	-	-	-	-	-	-	-	-	0.715

Table 3.7: Experimental results obtained by applying $CVSimP$. A comparison to previous work is depicted

We used a mini batch stochastic gradient descent approach enhanced with Adam optimizer [KB14] using a batch size of 32. We set the initial learning rate to $1e-3$ and use a reduction policy when no improvement is detected for 5 epochs. In this case the learning rate is reduced by 50%. The best performing model on validation set is kept.

Results and discussion

In Table 3.7 we reported the performance for each protein. The corresponding superfamily (as denoted in Table 3.6) is also marked. The overall $F1$ weighted score obtained is 0.867.

For comparison against previous work we have considered two approaches that we recently proposed and tested on the same set of proteins. The approaches were based on autoencoders [TCC19] and clustering [TCB19].

3.4.3 Conclusions and future work

We presented in this work an approach $CVSimP$ based on *siamese convolutional neural networks* for predicting proteins structural similarity using protein images obtained from the Protein data bank [BWF+00]. A dataset consisting of graphical representations of 57 proteins belonging to 9 superfamilies was used. We have developed an one-shot learning approach for which encouraging results were obtained (an overall *F-measure* above 0.867 has been obtained for the binary classification task).

Final Conclusions and Future Work

Final Conclusions

In this thesis we have investigated the problem of analyzing the conformational transitions of proteins, with the more general goal of contributing to a comprehensive understanding of the problem. We presented the current state-of-the-art approaches and we proposed some new computational perspectives on the problem, based on *machine learning*.

Another research direction that we pursued in our study was computer vision. The main goal was to develop deep learning based methods for tackling several real world problems.

We argued that although these two domains are distinct, the complexity that characterizes them may be a touching point. Therefore, our main effort was oriented towards leveraging the effectiveness of deep learning techniques for developing state of the art methods.

From a bioinformatics perspective, we also highlighted, through several experiments, that the information obtained through analyzing proteins conformational transitions capture the relationships between related proteins, relations which are confirmed from a biological perspective.

Moreover, we have conducted a study towards the application of *unsupervised machine learning* methods for analyzing protein conformational transitions in order to extract information about their structural similarity. A study towards applying *deep autoencoders* was also developed in order to give a better comprehension of protein dynamics. The experiments conducted showed that *autoencoders* are effective unsupervised models able to learn the structure of proteins, as well as to uncover the structural similarity between proteins. Moreover, we obtained empirical evidence that autoencoders are able to encode hidden patterns relevant from a biological perspective. The approach was extended for inter proteins analysis using an ensemble of deep autoencoders.

Clustering was also used as an *unsupervised classification* method for investigating the relevance of RSA values for predicting internal transitions of proteins. The method was also used for finding similarities between proteins. *Principal component analysis* was explored for data visualization and used to examine how RSA values evolve between conformational transitions. The experiments conducted on several proteins highlighted that RSA values are smoothly changing between conformational transitions.

On the computer vision side we investigated various problems. We tackled the image classification problem on two particular tasks that were tackled using *deep convolutional neural networks*. Thus, we presented our work for signature authenticity analysis. Moreover, we developed a state of the art technique from both a classification performance and an efficiency perspective on an open source dataset for fruits classification. We also discussed another task, namely document localization. A method for this problem was proposed with the broader goal to create a document deskewing pipeline. An interdisciplinary study that applied computer vision techniques for studying proteins was finally discussed.

Future work

We plan to extend the analysis of the results obtained in this thesis from a biological viewpoint. Based on this study we aim to advance our research towards predicting protein conformational transitions. In addition, future work will be carried out in order to combine RSA values with Structural alphabet representations [PFK10] of protein transitions in order to gain further insight into protein conformational transitions.

On the computer vision side we plan to continue improving our current performance on the discussed tasks. Moreover, we plan to continue our investigations for applying similar techniques for new practical domains such as health care and environmental related problems such as waste management and recycling.

Bibliography

[ACT18] Silvana Albert, Gabriela Czibula, and Mihai Teletin. Analyzing the impact of protein representation on mining structural patterns from protein data. In *2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 000533–000538, May 2018.

[AJW18] Joel Andersson, Eskil Jarlskog, and Richard Wang. Fruit recognition. *Report, University of California San Diego*, 2018.

[ATC18] Silvana Albert, Mihai Teletin, and Gabriela Czibula. Analysing protein data using unsupervised learning techniques. *International Journal of Innovative Computing, Information and Control*, 14:861–880, 2018.

[Bar18] Mateusz Baryła. What is this fruit? neural network application for vietnamese fruit recognition. In *ITM Web of Conferences*, volume 20, page 02009. EDP Sciences, 2018.

[BBCG13] Jacques M Bahi, Wojciech Bienia, Nathalie Côté, and Christophe Guyeux. Is protein folding problem really a np-complete one? first investigations. *arXiv preprint arXiv:1306.1372*, 2013.

[BCB94] Daniel Pierre Bovet, Pierluigi Crescenzi, and D Bovet. *Introduction to the Theory of Complexity*. Prentice Hall London, 1994.

[BDF+15] Guillaume Bouvier, Nathan Desdouits, Mathias Ferber, Arnaud Blondel, and Michael Nilges. An automatic tool to analyze and cluster macromolecular conformations based on self-organizing maps. *Bioinformatics*, 31(9):1490–1492, 2015.

[BGL+94] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. In *Advances in neural information processing systems*, pages 737–744, 1994.

[BHN04] Helen Berman, Kim Henrick, and Haruki Nakamura. Berman, h, henrick, k and nakamura, h. announcing the worldwide protein data bank. nat struct biol 10: 980. *Nature structural biology*, 10:980, 01 2004.

[BL98] BONNIE BERGER and TOM LEIGHTON. Protein folding in the hydrophobic-hydrophilic (hp) model is np-complete. *Journal of Computational Biology*, 5(1):27–40, 1998. PMID: 9541869.

[BPC+17] Maria-Iuliana Bocicor, Alesandro Pandini, Gabriela Czibula, Silvana Albert, and Mihai Teletin. Using Computational Intelligence Models for Additional Insight into Protein Structure. *Studia Universitatis "Babeş-Bolyai" Informatica*, 62:107–119, 2017.

[BWF+00] H.M Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The Protein Data Bank. *Nucleic Acids Res.*, 28:235–242, 2000.

[C+15] F. Chollet et al. Keras. https://github.com/fchollet/keras, 2015.

[CCT19] Gabriela Czibula, Carmina Codre, and Mihai Teletin. *AnomalP*: A new approach for detecting anomalous protein conformations using deep autoencoders. *Expert systems with applications*, page under review, 2019.

[Cho16] François Chollet. Xception: Deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357, 2016.

[DDS+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition.*, pages 248–255. IEEE, 2009.

[DT19] Lorand Dobai and Mihai Teletin. A document detection technique using convolutional neural networks for optical character recognition systems. *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning 2019, Bruges, Belgium*, pages 547–552, 2019.

[GBC16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[HCL06] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[HZC+17] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[ICT17] Vlad-Sebastian Ionescu, Gabriela Czibula, and Mihai Teletin. Supervised learning techniques for body mass estimation in bioarchaeology. In *IEEE 7th International Workshop on Soft Computing Applications (SOFA 2016)*, pages 71–86. Springer, 2017.

[ITV16] Vlad-Sebastian Ionescu, Mihai Teletin, and Estera-Maria Voiculescu. Machine learning techniques for age at death estimation from long bone lengths. In *IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI 2016)*, pages 457 – 462. IEEE Hungary Section, 2016.

[JS17] Khurram Javed and Faisal Shafait. Real-time document localization in natural images by recursive application of a cnn. In *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, volume 1, pages 105–110. IEEE, 2017.

[KB14] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[KC10a] Bence Kovari and Hassan Charaf. Analysis of intra-person variability of features for off-line signature verification. *W. Trans. on Comp.*, 9(11):1359–1368, November 2010.

[KC10b] Bence Kovari and Hassan Charaf. Statistical analysis of signature features with respect to applicability in off-line signature verification. In *Proceedings of the 14th WSEAS International Conference on Computers: Part of the 14th WSEAS CSCC Multiconference - Volume II*, ICCOMP'10, pages 473–478, Stevens Point, Wisconsin, USA, 2010. World Scientific and Engineering Academy and Society (WSEAS).

[KC13] Bence Kovari and Hassan Charaf. A study on the consistency and significance of local features in off-line signature verification. *Pattern Recognition Letters*, https://www.aut.bme.hu/Pages/Research/Signature/Resources, 2013.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[KU03] Daniel Keysers and Walter Unger. Elastic image matching is np-complete. *Pattern Recogn. Lett.*, 24(1):445–453, January 2003.

[LCY13] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *CoRR*, abs/1312.4400, 2013.

[LJN06] D.P. Lewis, T. Jebara, and W. S. Noble. Support vector machine learning from heterogeneous data: an empirical analysis using protein sequence and structure. *Bioinformatics*, 22(22):27532760, 2006.

[MDH+10] T. Meyer, M. D'Abramo, A. Hospital, M. Rueda, C. Ferrer-Costa, A. Prez, O. Carrillo, J. Camps, C. Fenollosa, D. Repchevsky, J.L. Gelp, and M. Orozco. MoDEL: A database of atomistic molecular dynamics trajectories. *Structure*, 18(11):1399 – 1409, 2010.

[MJC02] K. K. Moon, R. L. Jernigan, and G. S. Chirikjian. Efficient generation of feasible pathways for protein conformational transitions. *Biophysical Journal*, 83(3):1620–1630, 2002.

[MMC08] G. Morra, M. Meli, and G. Colombo. Molecular dynamics simulations of proteins and peptides: from folding to drug design. *Current Protein and Peptide Science*, 9:2181–2196, 2008.

[MO18] Horea Mureşan and Mihai Oltean. Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica*, 10(1):26–42, 2018.

[MZZS18] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. *arXiv preprint arXiv:1807.11164*, 2018.

[PFK10] A. Pandini, A. Fornili, and J. Kleinjung. Structural alphabets derived from attractors in conformational space. *BMC Bioinformatics*, 11(97):1–18, 2010.

[PGCC17] Adam Paszke, Sam Gross, Soumith Chintala, and Gregory Chanan. Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration. https://github.com/pytorch/pytorch, 2017.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[RBV+18] Alexander S Rose, Anthony R Bradley, Yana Valasatava, Jose M Duarte, Andreas Prlić, and Peter W Rose. Ngl viewer: web-based molecular graphics for large complexes. *Bioinformatics*, 34(21):3755–3758, 2018.

[RGSK11] Bernardete Ribeiro, Ivo Gonçalves, Sérgio Santos, and Alexander Kovacec. Deep learning networks for off-line handwritten signature recognition. In César San Martin and Sang-Woon Kim, editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 523–532, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[SEZ+13] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[SHZ+18] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.

[SSB06] Harish Srinivasan, Sargur N. Srihari, and Matthew J. Beal. Machine learning for signature verification. In Prem K. Kalra and Shmuel Peleg, editors, *Computer Vision, Graphics and Image Processing*, pages 761–775, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[SVI+15] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015.

[TC20] Mihai Teletin and Gabriela Czibula. Cvsimp: An approach for predicting proteins' structural similarity using one-shot learning. In *2020 IEEE 14th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, volume in press, 2020.

[TCAB18] Mihai Teletin, Gabriela Czibula, Silvana Albert, and Mariana-Iuliana Bocicor. Using unsupervised learning methods for enhancing protein structure insight. *Procedia Computer Science*, 126:19 – 28, 2018. Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 22nd International Conference, KES-2018, Belgrade, Serbia.

[TCB+18] Mihai Teletin, Gabriela Czibula, Mariana-Iuliana Bocicor, Silvana Albert, and Alessandro Pandini. Deep autoencoders for additional insight into protein dynamics. In *Artificial Neural Networks and Machine Learning – ICANN 2018*, pages 79–89, Cham, 2018. Springer International Publishing.

[TCB19] Mihai Teletin, Gabriela Czibula, and Maria-Iuliana Bocicor. Using clustering models for uncovering proteins' structural similarity. In *2019 IEEE 13th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, volume in press, pages 185–190, 2019.

[TCC19] Mihai Teletin, Gabriela Czibula, and Carmina Codre. Autosimp: An approach for predicting proteins structural similarities using an ensemble of deep autoencoders. In *International Conference on Knowledge Science, Engineering and Management*, pages 49–54. Springer, 2019.

[TD19] Mihai Teletin and Lorand Dobai. Lightweight models for fruits recognition. In *2019 IEEE 13th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, volume in press, pages 69–74, 2019.

[Tel17] Mihai Teletin. Machine Learning Techniques for Detecting False Signatures. *Studia Universitatis "Babeş-Bolyai" Informatica*, 62:49–59, 2017.

[TT09] Nobuhiko Tokuriki and Dan S. Tawfik. Protein dynamism and evolvability. *Science*, 324(9524):203–207, 2009.

[Xio16] Ying Xiong. Fast and accurate document detection for scanning, August 2016.

[YG04] Y. Ye and A. Godzik. FATCAT: a web server for flexible structure comparison and structure similarity searching. *Nucleic Acids Res.*, 32:582–585, 2004.