Generating Nature-Resembling Tertiary Protein Structures with Advanced Generative Adversarial Networks (GANs)

Mena Nagy A. Khalaf[1], Taysir Hassan A Soliman[2], Sara Salah Mohamed[3]
Information System Department-Faculty of Computer and Information, Assiut University, Assiut, 71515, Egypt[1,2]
Mathematics and Computer Science Department-Faculty of Science, New Valley University, New Valley, 72713, Egypt[3]

***Abstract*—In the field of molecular chemistry, the functions, interactions, and bonds between proteins depend on their tertiary structures. Proteins naturally exhibit dynamism under different physiological conditions, as they alter their tertiary structures to accommodate interactions with other molecular partners. Significant advancements in Generative Adversarial Networks (GANs) have been leveraged to generate tertiary structures closely mimicking the natural features of real proteins, including the backbone and local and distal characteristics. Our research has led to the development of stable model ROD-WGAN, which is capable of generating tertiary structures that closely resemble those found in nature. Four key contributions have been made to achieve this goal: (1) Utilizing Ratio Of Distribution (ROD) as a penalty function in the Wasserstein Generative Adversarial Networks (WGAN), (2) Developing a GAN network architecture that fertilizes the residual block in generator, (3) Increasing the length of the generated protein structures to 256 amino acids, and (4) Revealing consistent correlations through Structural Similarity Index Measure (SSIM) in protein structures with varying lengths. These model represent a significant step towards robust deep-generation models that can explore the highly diverse set of protein molecule structures that support various cellular activities. Moreover, they provide a valuable source of data augmentation for critical applications such as molecular structure prediction, inpainting, dynamics, and drug design. Data, code, and trained models are available at https://github.com/mena01/Generating-Tertiary-Protein-Structures-Resembling-Nature-using-Advanced-WGAN.**

***Keywords*—Molecular structure; protein structure; protein modeling; tertiary structure; generative adversarial learning; deep learning; proteomic**

I. Introduction

Molecular structures have been extensively researched over the past century due to their significant impact on our understanding of the human body and its functioning, both in normal and pathological states. This has facilitated the identification of the molecular basis of various diseases and facilitated the development of new strategies for their prevention and treatment [1]. In recent years, the pivotal role of bioinformatics models in the analysis of the molecular basis of diseases, including infectious diseases and cancers such as gallbladder cancer [2], lung cancer [3], colon cancer [4], [5], and prostate cancer [6], has been increasingly recognized.

The function and interactions of molecules largely depend on their structure. Therefore, predicting the structure of molecules can provide insights into their functions and has implications for a wide range of applications, including drug design [7], molecule structure prediction [8], molecular inpainting [9], and molecular dynamics [10].

There are four different structures that proteins can have: primary structures [11], secondary structures [12], tertiary structures [13], and quaternary structures [14]. In biological laboratories, there are traditional methods that are used to determine these protein structures, such as X-ray crystallography [15], nuclear magnetic resonance (NMR) [16], and cryogenic electron microscopy (cryo-EM) [17]. However, these methods can be time-consuming and resource-intensive.

The gap between the number of known protein sequences and the number of discovered tertiary structures has increased exponentially and continues to grow [18]. According to the Protein Data Bank (PDB) [19], only around 180 thousand protein structures have been identified, compared to the approximately 207 million known protein sequences according to Uniport/TrEMBL [20].As data scientists working in the field of protein structure prediction, our role is to generate tertiary structures of proteins that accurately mimic natural protein structures by capturing the natural protein structures' distribution.

CASP (Critical Assessment of protein Structure Prediction) [21] evaluates models that predict protein structures, and the recent introduction of Google's DeepMind AlphaFold v2 [22] has achieved the greatest performance in this area. It is important to note that proteins are naturally dynamic molecules [23] that can adopt different tertiary structures to modulate their interactions with different partners.

The dynamics of proteins have garnered significant attention lately, as evidenced by recent studies [24], [25], [26] that examine the balance motions between the spike glycoprotein (Receptor-Binding Domain (RBD) of the severe acute respiratory syndrome coronavirus 2 (SARS-COV-2)) and the human Angiotensin-converting enzyme 2 (ACE2) receptor.

The spike glycoprotein is flexible and can transition between a closed and partially open structure, allowing it to bind to the ACE2 receptor and act as a viral entry point into human host cells.

Therefore, it is important to detect the diverse protein structures that proteins can access to regulate interactions with their molecular partners. Obtaining a broad view of the structure space is thus a vitally important research problem, and much work [26] has been focused on modeling proteins to capture this broad view of the protein structure space.

However, this is a challenging task, and most research [27] relies on existing protein structure data or restricted physical models [28] to guide search algorithms to the pertinent regions of the structure space that are otherwise too vast [29].

Early models used angles between bonds of atoms to simulate protein structures [9], [30] but more recent work has used GANs and long short-term memory networks to generate protein structures based on alpha-Carpon [30]. Despite the promising results obtained from these models, there is still much work to accurately simulate the diversity of protein structure.

In [9], the researchers used GANs with backbone angles in the representation of tertiary proteins, but they expanded the training dataset to include more proteins with various structures. However, it was observed that the generated protein structures exhibited distortion. As a result, the researchers replaced the backbone angles with distance matrices, which incorporated either the distances between each pair of Carbon Alpha (CA) atoms in the protein's main chain or the distances between every atom in the protein [31]. In the latter, the number of atoms increases, leading to larger distance matrices that can be difficult and time-consuming to train.

Recently, GAN networks have been employed to predict contact maps for protein structures [32], [33]. In this context, a contact map is a matrix in which the value of each element is 1 if two CA amino acids are in contact and 0 otherwise.

In [34], researchers trained their autoencoder (AC) on structures obtained from molecular dynamics simulations, such as computational platforms. In [35], the researchers used Rosetta as a platform for protein structure prediction to train the AC of Variational Autoencoder (VAE) [36]. In both cases, the researchers did not use experimental protein structures from the Protein Data Bank (PDB). However, in GAN models, it is preferable to use experimental structures from PDB rather than computational platforms.

In [10], the author used distance matices of CA and produced nine models based on Vanilla GANs, which include Vanilla GAN, vanilla GAN + TTUR, Vanilla GAN + SpecNorm, Vanilla GAN + VBN, Vanilla GAN + TTUR + SpecNorm, Vanilla GAN + TTUR + VBN, Vanilla GAN + SpecNorm + VBN, Vanilla GAN + TTUR + SpecNorm + VBN, and WGAN.

The model achieved the highest accuracy was WGAN, denoted here as $WGAN_{Rahman}$, but it did not accurately capture the backbone and exhibited poor accuracy in both short-range and long-range structures. Furthermore, the generated distribution deviated significantly from the natural distribution, Where the average peptide bond lengths of $WGAN_{Rahman}$ at 128 amino acids for backbone, short-range, and long-range structures were 7.5 Å, 11.66 Å, and 26.144 Å, respectively. In comparison, the natural average peptide bond lengths for backbone, short-range, and long-range structures are 3.78 Å, 7.79 Å, and 21.3 Å, respectively.

In this paper, our objective is to create models using WGAN [37] to generate tertiary protein structures that exhibit similar features to the natural protein structures in terms of their backbone, local, and distal protein structures. Additionally, we aim to ensure that the distribution of the generated tertiary protein structures is comparable to that of the real tertiary protein structures.

We represented the tertiary structure using a CA distance matrix, as described in [9], [10]. Our models were trained using data from the PDB [19], which contains a diverse set of protein structures with varying amino acid lengths. We increased the amino acid length in our models to 256 aa. Additionally, we adjusted the WGAN gradient penalty by incorporating the ratio of distribution that achieved high accuracy within only 10 epochs. This contrasts with the best of the previous methods, where the $WGAN_{Rahman}$ model [19] was found to be unstable and achieved acceptable accuracy only after 50 epochs. To enhance stability, we utilized residual blocks in the generator network.

To summarise, the main contributions of our model ROD-WGAN, which make it different from the other models, are as follows:

1) Enhancing the WGAN gradient penalty by introducing the Ratio of Distribution (ROD) concept
2) Incorporating the convolution layers and the residual blocks in the Generator network to generate superior tertiary protein structures.
3) Increasing the length of the generated protein to 256 aa.
4) Our research reveals consistent correlations in protein structures through the application of Structural Similarity Index Measure (SSIM). These findings provide valuable insights into the inherent relationships within protein structures.

In the subsequent sections, this paper embarks on a comprehensive journey through the foundational elements of our study. The groundwork is established in Section II, where we present our proposed methodology and its key components. Progressing further, Section III meticulously details the refinement and preprocessing of our training dataset. Moving to Section IV, a thorough evaluation of our models takes place, wherein we compare them to state-of-the-art counterparts. Subsequent sections delve into the interpretation of experimental outcomes in Section V, while our contributions are summarized, and potential avenues for future research are suggested in Section VI, concluding this paper.

II. Proposed Methodology

The Generative Adversarial Network (GAN) [38] is a sophisticated architecture that has garnered attention from researchers across various fields, particularly in computer vision [39], [40], [41]. GAN has been employed to generate tertiary protein structures that mimic the real tertiary protein structure. In fact, this process is even more daunting than generating images due to the various constraints involved in the protein's structure, such as the backbone and short- and long-distance features. Previous GAN models have fallen short in capturing all three features of the tertiary protein structure with the same level of accuracy, and the discrepancy between the generated and the natural distributions was not close enough. The subsequent sections will briefly introduce the GAN model architecture and explain our model, ROD-WGAN.

A. GAN

GAN [38] consists of two neural networks that compete with each other: the Generator (G) and the Discriminator (D). The generator is responsible for generating fake proteins that simulate natural proteins and aims to deceive the discriminator, while the discriminator distinguishes between fake and real proteins.

As they compete against each other, each network tries to outperform the other. The balance between G and D leads to an optimal state in which their loss is equal to 0.5. Mathematically, assuming x represents the real data and z represents the latent vector or noise data, G is the generator that minimizes the function expressed in Eq. (1), and D is the discriminator that maximizes it.

$$\min_G \max_D GAN(G, D) = E_{xp_r(x)}[\log D(x)] \\ +E_{zp_z(z)}[\log 1 - D(G(z))] \tag{1}$$

Where p_r denotes the real data distribution, p_z denotes the model distribution, z is the input to the Generator and is randomly selected from some simple noise distribution.

The GAN network has encountered many problems, the most important of which are vanishing gradients and network instability. In [37], researchers proposed a WGAN network that uses Wasserstein distance to make the network more stable and faster, avoiding many of the issues faced with the GAN. The WGAN harnesses the 1-Lipschitz function, which guarantees the value is generated in a specific space and is enforced by the gradient penalty. It also replaces the name of the discriminator with the critic. The WGAN loss function is shown in Eq. (2) as follows:

$$L = \underbrace{E_{\widetilde{x} \sim p_g[D(x)]} - E_{xp_r}[D(x)]}_{originalcriticloss} + \underbrace{\lambda E_{\widehat{x} \sim p_{\widehat{x}}}[(\| \nabla_{\widehat{x}} D(\widehat{x}) \|_2 - 1)^2]}_{gradientpenalty} \tag{2}$$

Where $\widehat{x}$ is composed of real data x and fake data $\widetilde{x}$, which is defined as $\widetilde{x} = G(z)$, using the following equation:

$$\widehat{x} = \varepsilon x + (1 - \varepsilon)\widetilde{x} \tag{3}$$

B. Ratio of Distribution (ROD)

According to our experimental findings, we found that the sum of the values of each distance matrix remains largely consistent across different proteins with the same number of amino acids. For example, as shown in Fig. 1, the sum of values of the distance matrix of different proteins with a length of 128 aa is 375000 Angstrom Å.

To compute ROD, some steps are required:

1) Calculate the mean sum of the natural proteins' distances matrices with the same length on all batches denoted as μ_r, (only performed once).
2) Calculate the mean sum of the distance matrices of generated proteins with the same length for each batch, denoted as μ_f (performed every time the fake data is generated).

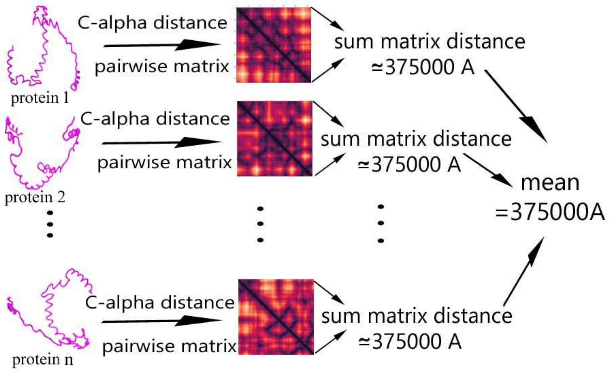


Fig. 1: Proteins with equal lengths of amino acids have equal sums of CA pairwise distance matrices.

3) Calculate the ratio of distribution ρ by dividing μ_r over μ_f.
4) Modify Equation (3) by adding the ratio of distribution ρ, as follows:

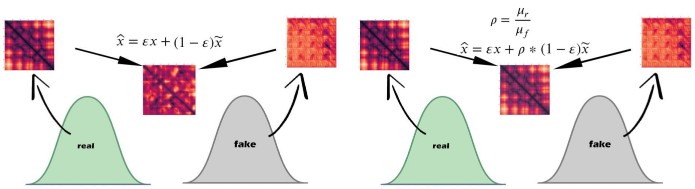
$$\rho = \frac{\mu_r}{\mu_f} \\ \widehat{x} = \varepsilon x + \rho * (1 - \varepsilon)\widetilde{x} \tag{4}$$

ROD ρ helped to generate close-to-real protein structures by capturing the backbone, short-range, and long-range features. In addition, the generated protein distribution is close enough to the real protein distribution, which accelerates and guarantees the stability of the learning process.

When μ_f is greater than μ_r, ρ is less than 1. Thus, we multiply the μ_f with the ρ to ensures that the mixed distance matrix value does not surpass the natural distance matrix value.

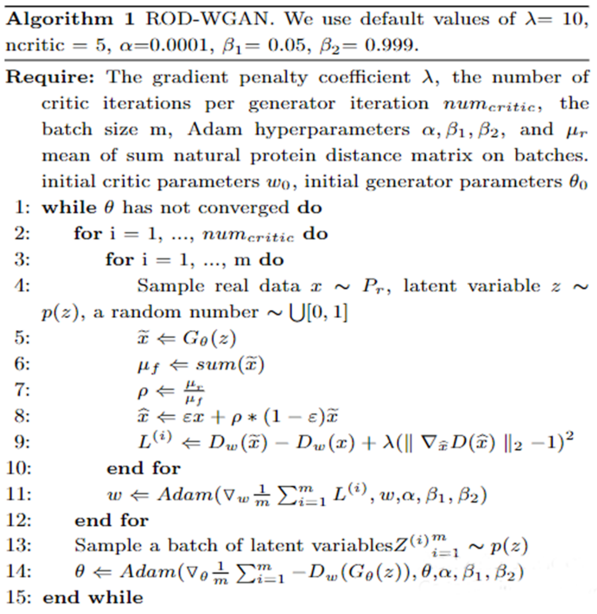
Conversely, when μ_f is smaller than μ_r, ρ is greater than 1. Thus, we multiply the μ_f with the ρ to ensures that the mixed distance matrix value does not fall below the natural distance matrix value.

In general, ρ controls the mixed distance matrix value to be aligned closely with the natural distance matrix value, as depicted in Fig. 2. The algorithm's steps are illustrated in Fig. 3.


$\widehat{x} = \varepsilon x + (1 - \varepsilon)\widetilde{x}$
real
fake
$\rho = \frac{\mu_r}{\mu_f}$
$\widehat{x} = \varepsilon x + \rho * (1 - \varepsilon)\widetilde{x}$
real
fake

(a) (b)

Fig. 2: The fake distribution trying to move to the real distribution by using a mixed distance matrix. a) Without the ratio of distribution and b) Using the ratio of distribution.



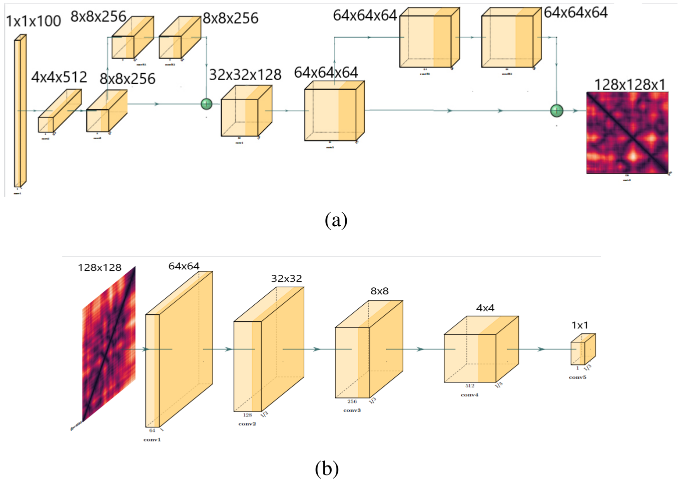
```
Algorithm 1 ROD-WGAN. We use default values of $\lambda$= 10, ncritic = 5, $\alpha$=0.0001, $\beta_1$= 0.05, $\beta_2$= 0.999.
Require: The gradient penalty coefficient $\lambda$, the number of critic iterations per generator iteration $num_{critic}$, the batch size m, Adam hyperparameters $\alpha, \beta_1, \beta_2$, and $\mu_r$ mean of sum natural protein distance matrix on batches. initial critic parameters $w_0$, initial generator parameters $\theta_0$
while $\theta$ has not converged do
    for i = 1, ..., $num_{critic}$ do
        for i = 1, ..., m do
            Sample real data $x \sim P_r$, latent variable $z \sim p(z)$, a random number $\sim \bigcup[0, 1]$
            $\tilde{x} \Leftarrow G_\theta(z)$
            $\mu_f \Leftarrow sum(\tilde{x})$
            $\rho \Leftarrow \frac{\mu_r}{\mu_f}$
            $\hat{x} \Leftarrow \varepsilon x + \rho * (1 - \varepsilon)\tilde{x}$
            $L^{(i)} \Leftarrow D_w(\tilde{x}) - D_w(x) + \lambda(\| \nabla_{\hat{x}} D(\hat{x}) \|_2 - 1)^2$
        end for
        $w \Leftarrow Adam(\nabla_w \frac{1}{m} \sum_{i=1}^{m} L^{(i)}, w, \alpha, \beta_1, \beta_2)$
    end for
    Sample a batch of latent variables$Z^{(i)}{}_{i=1}^{m} \sim p(z)$
    $\theta \Leftarrow Adam(\nabla_\theta \frac{1}{m} \sum_{i=1}^{m} -D_w(G_\theta(z)), \theta, \alpha, \beta_1, \beta_2)$
end while
```

Fig. 3: The ROD-WGAN algorithm.

C. Model Architecture

The model architecture of GAN consists of two networks: the generator network and the discriminator network.

1) The Generator Network Architecture: The generator architecture of our models is illustrated in Fig. 4(a). The G network consists of convolution layers, which are fertilized by two residual blocks to enhance and accelerate the model's learning. Table I provides the specifics of the generator network parameters for the 128 aa.



(a)

(b)

Fig. 4: The proposed architecture ROD WGAN a) Represents the Generator which consists of two Residual blocks and b) Represents the discriminator.

2) The discriminator network architecture: Fig. 4(b) shows the architecture of the discriminator. The discriminator takes the distance matrix of the protein structure produced by G and the distance matrix of the natural protein structure as input to differentiate the natural matrix from the generated one. The discriminator utilizes five convolution layers. Table II shows the discriminator network parameters for 128 aa .

TABLE I: The Layers of the Generator Network Architecture

layer		Details	filter	stride	padding
input		100*1*1	-	-	-
conv1		512*4*4	512*4*4	4	0
conv2		256*8*8	256*4*4	2	1
Residual1	conv	256*8*8	256*3*3	1	1
	conv	256*8*8	256*3*3	1	1
conv3		128*32*32	128*4*4	4	0
conv4		64*64*64	64*4*4	2	1
Residual2	conv	64*64*64	64*3*3	1	1
	conv	64*64*64	64*3*3	1	1
conv5		1*128*128	1*4*4	2	1

TABLE II: Layers of the Discriminator Network Architecture

layer	Details	filter	stride	padding
input	1*128*128	-	-	-
conv1	64*64*64	64*4*4	4	0
conv2	128*32*32	128*4*4	2	1
conv3	256*8*8	256*4*4	4	0
conv4	512*4*4	512*4*4	2	1
conv5	1*1*1	1*1*1	2	1

III. Training Dataset

The dataset utilized comprised 115K protein structures sourced from PDB [19], with variations in there protein size. We calculated distance matrices that measured the distance between each pair of CA atoms within the protein's main chain. As a result, the matrix distance size equaled n*n, where n equaled 64, 128, or 256 aa. It's noteworthy that we are the first to create the 256aa structure; as the number of amino acids increases, the size of the distance matrix also increases, thereby increasing the complexity of model training.

To the extent of our knowledge, there has been no prior study on the generation of protein structures comprising 256 aa using the generative models. Our results align with recent studies on protein design via deep learning techniques [42], [43], as well as the latest advancements in the field of protein structure prediction and design [8], [9], [10].

IV. Assessment of Our Models

To evaluate the performance of our models as well as other state-of-the-art methods, we conducted several assessments, including i) Quantitative assessment, which involved evaluating the average peptide bond and comparing distributions; ii) Qualitative assessment; and iii) Convergence analysis.

A. Assessment on the Average Peptide Bond

The average peptide bond is calculated by summing all the entries along the main diagonal of the distance matrix and then dividing that sum by the length of the diagonal. To assess the quality of the generated protein tertiary structure, we compared its features (distance of backbone, short-range, and long-range) to those of natural proteins.

1) Assessment on the backbone structure: The backbone refers to the main diagonal of the generated distance matrix, which is constructed using every consecutive (i, i+1) CA pair where 0<i<n-1. In a natural protein, the ideal distance between two consecutive amino acids is 3.79Å

2) Assessment on the short-range structure and the long-range structure: After computing the backbone, we can calculate the local structures by examining the short-range distance between consecutive (i, i+j) CA pairs, where j is between 1 and 4. In natural proteins, the ideal short-range distance is 7.8Å. If we increase j beyond 4, we can determine the long-range distance, or distal structure. For 64 aa in a natural protein, the ideal long-range distance is 18.31Å. While for 128 aa in a natural protein, the ideal long-range distance is 21.31Å. Finally, when we computed it for 256 aa, we found a value of 25.01Å, based on experimental data obtained from natural proteins.

B. Structural Similarity Index Measure

In our study and during our experiments, we made a noteworthy observation regarding the tertiary protein structures: there exists a consistent correlation between natural protein structures that have the same number of amino acids. When we calculate the SSIM between two different natural distance matrices, we obtained the following constant values for different lengths of distance matrices: 0.72 for distance matrices with a length of 64 aa, 0.69 for distance matrices with a length of 128 aa, and 0.68 for distance matrices with a length of 256 aa.

Based on these findings, we utilized SSIM as a loss function to enhance the similarity and correlation between the natural and the generated tertiary protein structures. We evaluated the SSIM score between the natural and the generated structures using Eq. (6).

$$SSIM_{(fake,real)} = \frac{2\mu_{real}\mu_{fake}+C_1}{\mu_{real}^2\mu_{fake}^2+C_1} * \frac{2\sigma_{real}\sigma_{fake}+C_2}{\sigma_{real}^2\sigma_{fake}^2+C_2} * \frac{2\sigma_{real*fake}+C_3}{\sigma_{real}\sigma_{fake}+C_3} \quad (5)$$

The formula includes several variables, such as the mean values of the real and the fake protein (μ_{real} and μ_{fake}, respectively), the standard deviation of the real and the fake protein (σ_{real} and σ_{fake}, respectively), as well as the cross-correlation ($\sigma_{real}\sigma_{fake}$) between the two proteins. Additionally, the formula contains three constants, labeled C_1, C_2, and C_3 equal to 0.01, 0.03, and 0.015, respectively [44].

The SSIM score ranges from 0 to 1, and a score closer to 1 indicates a greater level of correlation between the real and fake images, and vice versa. Therefore, we strive to achieve an SSIM score that is as close to natural as possible. Based on our experiments, if we calculate the SSIM between a natural distance matrix (n*n), where 'n' represents the length of the protein (either 64, 128, or 256), and itself, the SSIM value will always be one.

For example, if we take natural protein1 with a length of 64 aa, and natural protein2 with a length of 64 aa, and calculate the SSIM between the distance matrices of these proteins, we will find the value to be 0.72. If we repeat the calculation for two different proteins, we will obtain the same constant value of 0.72. Hence, when calculating the SSIM between two different proteins with a length of 64 aa, the value is always constant at 0.72.

Similarly, if we take natural protein1 with a length of 128 aa, and natural protein2 with a length of 128 aa, and calculate the SSIM between the distance matrices of these proteins, we will find the value to be 0.69. If we repeat the calculation for different proteins, we will again obtain the same constant value of 0.69. Therefore, when calculating the SSIM between two different proteins with a length of 128 aa, the value is always constant at 0.69, regardless of the protein lengths.

Lastly, if we take natural protein1 with a length of 256 aa and natural protein2 with a length of 256 aa and calculate the SSIM between the distance matrices of these proteins, we will find the value to be 0.68. If we repeat the calculation for different proteins, we will once again obtain the same constant value of 0.68. Thus, when calculating the SSIM between two different proteins with a length of 256 aa, the value is always constant at 0.68.

C. Comparison of the Distribution

In GANs, we aim to capture the distribution of natural tertiary protein structures by approximating the generated distribution to the natural one. To measure the distance between the two distributions, we employ various metrics, such as the Earth Mover's Distance (EMD), Maximum Mean Discrepancy (MMD), and Bhattacharya Distance (BD).

1) Earth Mover's Distance (EMD): The Earth Mover's Distance, also known as the Wasserstein distance [45], represents the minimum cost required to transform the generated distribution of tertiary protein structures to the natural distribution. EMD has been found to provide better perceptual dissimilarity than any other dissimilarity measure. EMD measures the distance between the two distributions, where a lower EMD value indicates higher similarity or proximity between the distributions, and a higher EMD value indicates lower similarity.

2) Maximum Mean Discrepancy (MMD): Maximum Mean Discrepancy (MMD) [46] is a popular statistical test used to measure the distance between two distributions, p(A) and q(B). MMD is defined as the largest difference in the expectations of the mean of A(μ_A) and the mean of B(μ_B) over functions in the unit ball of a reproducing kernel Hilbert space (RKHS). MMD can be computed using Eq. (10). MMD measures the distance between the two distributions in the RKHS, where a lower MMD value indicates higher similarity or closeness between the distributions, and a higher MMD value indicates lower similarity.

$$MMD_{(A,B)} =| \mu_A - \mu_B |_H^2 \quad (6)$$

3) Bhattacharya distance (BD): Bhattacharya Distance (BD) [47] is another measure of the distance between two distributions p(a) and q(a) on the same domain. BD can be computed by Eq. (11).

$$BD_{(p,q)} = -\ln(BC(p,q)) \tag{7}$$

where the Bhattcharaya Coefficient BC is

$$BC_{(p,q)} = \sum_{x \in X} \sqrt{p(x)q(x)} \tag{8}$$

BC is an approximation that quantifies the degree of overlap between two samples drawn from distinct statistical distributions. A lower BD value indicates higher similarity or overlap between the two distributions, while a higher BD value indicates lower similarity.

V. Experimental Results and Discussions

We created ROD-WGAN model using the PyTorch framework on an RTX2080. We set the learning rate to 0.001 for both the critic and generator and used the Adam optimizer with b1 and b2 values of 0.5 and 0.999 respectively. The training time for one epoch of ROD-WGAN was approximately 17 minutes.

A. Quantitative Assessment

1) The effect of ROD: We have made significant progress in generating a distance matrix of tertiary protein structure using ROD. From the first ten epochs, we were able to capture the backbone, short-distance, and long-distance features of protein structures. As shown in Table III, our model, ROD-WGAN, outperformed WGAN without ROD and achieved better results that more closely resemble real protein structures. Furthermore, the distribution of the generated proteins is much closer to the natural protein distribution.

TABLE III: The Effect of ROD on the Results of Backbone, Short-Range, and Long-Range Protein Structures on Just 10 Epoch

Number of epoch	**Features**	**Natural**	**WGAN without ROD**	**ROD-WGAN**
10	Backbone	3.78	1.85	3.47
	Short	7.8	3.82	7.02
	Long	21.3	11.20	19.24

2) Average peptide bond: As mentioned earlier, we evaluated the distance matrix of the tertiary protein structure by considering the average length of peptide bonds in the backbone, short-range, and long-range distances. This method enabled us to accurately assess the similarity between the generated and natural distance matrices of the tertiary protein structure.

We assessed the quality of the backbone of the distance matrices generated by different models, namely ROD-WGAN, and $WGAN_{Rahman}$ [10]. As shown in Table IV and Fig. 5, we found that our model ROD-WGAN was able to capture the backbone, short-range, and long-range features of natural proteins more accurately than $WGAN_{Rahman}$ [10].

TABLE IV: Distance Features of the Backbone, the Short-Range, and the Long-Range for Natural and Generated Proteins by a Variety of Models

		Natural	**ROD-WGAN**	**WGAN$_{Rahman}$**
64aa	Backbone	3.78	**3.08**	5.05
	Short	7.5	**6.42**	9.43
	Long	17.55	**15.12**	20.11
128aa	Backbone	3.78	**3.014**	7.506
	Short	7.8	**6.58**	11.66
	Long	21.31	**19.24**	26.144
256aa	Backbone	3.78	**2.939**	-
	Short	7.55	**5.88**	-
	Long	25.01	**18.738**	-

**The bold characters indicate the best evaluation scores.

TABLE V: SSIM between the Natural and the Generated Distance Matrices by ROD-WGAN, and $WGAN_{Rahman}$

SSIM	**Natural**	**ROD-WGAN**	**WGAN$_{Rahman}$**
64aa	72.47%	73.79%	72.02%
128aa	69.60%	70.19%	66.74%
256aa	68.13%	69.63%	-

3) SSIM: As previously mentioned, SSIM is a metric used to assess the quality and similarity between two distance matrices. Table V displays the performance of ROD-WGAN, and $WGAN_{Rahman}$ [10] on distance matrices of 64 aa, 128 aa, and 256 aa.

The ROD-WGAN model provided the highest protein structural similarity distance matrices, with ROD-WGAN being closer to the natural than $WGAN_{Rahman}$, particularly as the number of amino acids increased.

4) Evaluation of the distribution distance: We employed a variety of measurements, including EMD, MMD, and BD, to assess the disparity between the distribution of the generated distance matrices for the tertiary protein structure and the distribution of the natural distance matrices for the tertiary protein structure.

The line graph in Fig. 6 illustrates the performance of our models during the training process. Specifically, the plot depicts the changes in EMD, BD, and MMD values over time for each model. The consistently lower lines for our model, ROD-WGAN, as compared to the best state-of-the-art model $WGAN_{Rahman}$, serve as evidence of the accuracy of our models in generating protein structures that closely resemble those found in nature.

Fig. 6 illustrates that ROD-WGAN outperform $WGAN_{Rahman}$ [10] in the 64aa and 128aa regions. Furthermore, We observed that the MMD, BD, and EMD values obtained for the 256 aa region closely resemble those of the natural protein. This is noteworthy as the $WGAN_{Rahman}$ model [10] was originally implemented for a limited region of 128 aa and does not cover the entire 256 aa. Overall, the ROD-WGAN model accurately capture the distribution of the natural protein.

B. Qualitative Assessment

In Fig. 7, we present the 64*64, 128*128, and 256*256 distance matrices for the generated tertiary protein structures of ROD-WGAN, and $WGAN_{Rahman}$ models and the natural

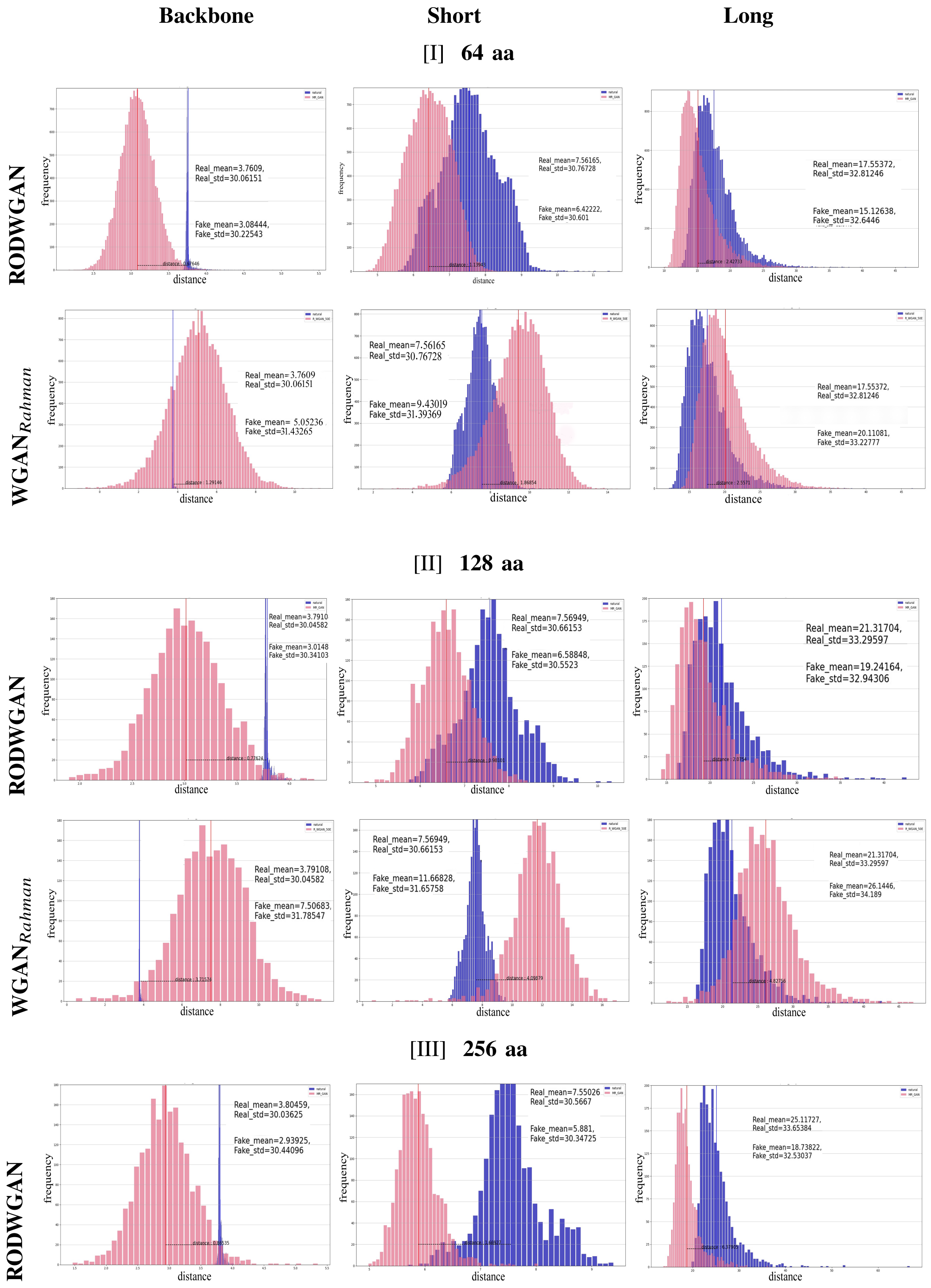


Fig. 5: The comparison has between the natural distribution(represented by the blue color)and the generated distribution(represented by the pink color). The $WGAN_{Rahman}$ model was not implemented on 256 aa.

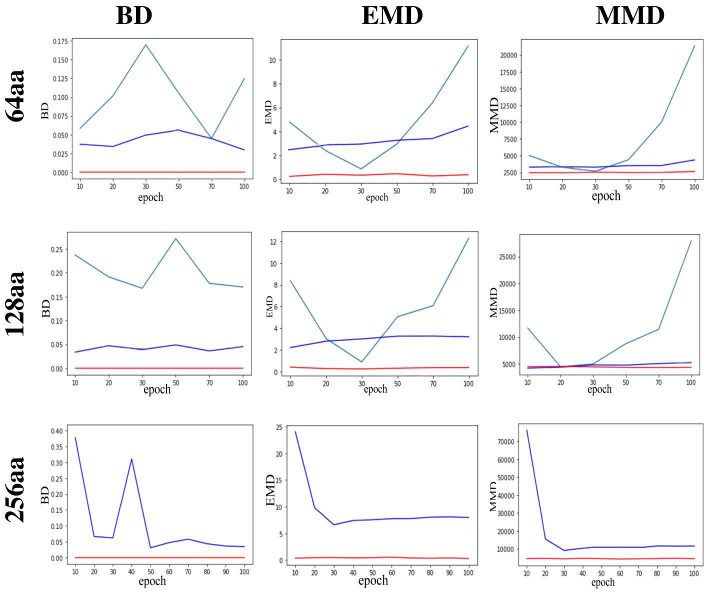


Fig. 6: Performance of ROD-WGAN, and $WGAN_{Rahman}$ on distributions for 64aa, 128aa, and 256aa.

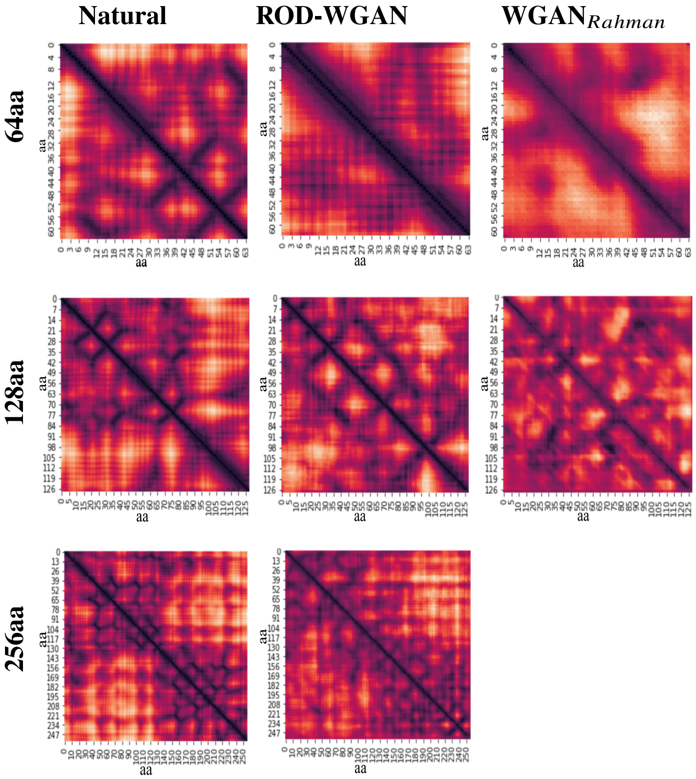


Fig. 7: Heatmaps visualized the Distance Matrices of the proteins' tertiary structures. The natural and generated distance matrices from various models ROD-WGAN, and $WGAN_{Rahman}$. The $WGAN_{Rahman}$ model [10] was not implemented on 256 aa.

structure. The heatmaps of these matrices were randomly selected from each model, with lighter colors indicating greater distance and darker colors indicating lower distance.

As seen in the 64aa and 128aa matrices, ROD-WGAN generated a clear heatmap distance matrix for the backbone, with a distinct dark diagonal, while the $WGAN_{Rahman}$ matrix was less clear. Furthermore, when we increase the length of the protein to 256aa, the ROD-WGAN model generate clear heat maps that have the backbone. To the best of our knowledge, there has been no previous report of generating protein structures with a length of 256 amino acids using the WGAN model. Our results are supported by recent surveys on protein design via deep learning [42], [43] and advances in protein structure prediction and design [8], [9], [10].

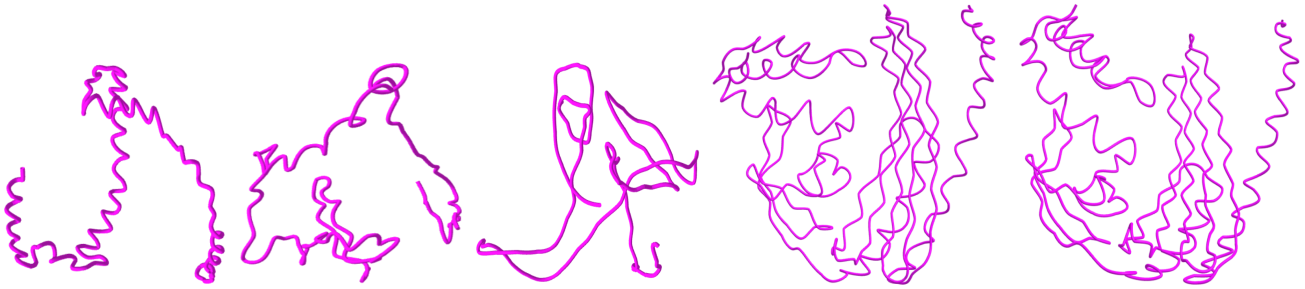
1) Alternating Direction Method of Multipliers (ADMM) : In our protein design study, we utilized the alternating direction method of multipliers (ADMM) [48] to convert the pairwise carbon alpha distance matrix (2d heatmap) to its equivalent 3d structure. We performed this for both the natural protein structures and those generated by various models ($WGAN_{Rahman}$, and ROD-WGAN). By employing the ADMM algorithm and implementing it with the software library [49], we were able to fold the distance matrices produced by our models to visualize the tertiary protein structures, as depicted in Fig. 8.

The visualization presented in Fig. 8 is crucial for evaluating the accuracy of the generated protein structures. It enables us to visually assess the overall shape of the generated structures and compare them against the natural structures. The ability to produce structures that closely resemble the natural structures is one of the most important characteristics of successful protein structure generation models. Therefore, the visualization in Fig. 8 provides an opportunity to validate the performance of our models in generating protein structures' distance matrices. We observed that the structures generated from our model ROD-WGAN was much closer to the natural protein structures compared to those generated from the $WGAN_{Rahman}$ model.

C. Convergence Analysis in ROD-WGAN Model for Protein Structure Generation

In this study, we investigated the effectiveness of the generator (G) and discriminator (D) in reducing loss and achieving convergence during training epochs while ensuring the generation of high-quality protein structures. Our focus was on the ROD-WGAN model, designed specifically for protein structure generation. Fig. 9 illustrates the performance of the ROD-WGAN model on datasets comprising varying lengths of amino acids (aa), namely 64 aa, 128 aa, and 256 aa. Our objective was to assess the model's convergence capability and loss reduction across these datasets.

The results demonstrated that the ROD-WGAN model outperformed in reducing the overall generator loss. This indicates the significant improvement of the generator network (G) in generating realistic protein structures as the training progressed. Furthermore, the convergence between the total generator loss and the critic loss exhibited by the model



(a) 128aa Natural (b) 128aa ROD-WGAN (c) 128aa WGAN$_{Rahman}$ model (d) 256aa natural (e) 256aa ROD-WGAN

Fig. 8: The tertiary structure of protein structures a) The natural protein structure with a length of 128aa b) The structure of a protein generated from ROD-WGAN with a length of 128aa. c) The structure of a protein generated from $WGAN_{Rahman}$ model [10] with a length of 128aa. d) The natural protein structure with a length of 256aa. e) The structure of a protein generated from ROD-WGAN with a length of 256aa.

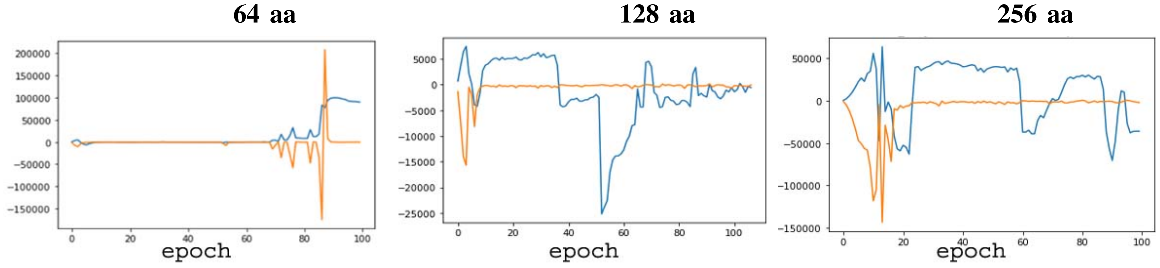


Fig. 9: Convergence analysis of ROD-WGAN losses.

indicated its stability and successful interplay between G and D.

VI. Conclusion and Future Work

In this study, we not only focused on predicting the protein tertiary structure problem, but we were also interested in making the method of prediction more simple, more practical, and less laborious. Despite the success of Alphafold in predicting the protein tertiary structure, there was still a need to search for another way that is easier, simpler, and does not require hundreds of TPUs (Tensor Processing Units).

We have developed models to generate distance matrices of proteins' tertiary structures in various amino acid lengths. Our proposed models are different from others in that they have the followings: 1) Modified the WGAN penalty equation by using the ROD 2) Developed Convolutional layers and enhanced it with the residual block 3) Applied on proteins with a length of 256 aa 4) our research uncovers consistent correlations in protein structures through the application of the SSIM. These findings provide valuable insights into the inherent relationships within protein structures, further enhancing the significance of our model.

In future work, we can try to generate tertiary protein structures based on distance and dihedral angle to increase the realism of the protein structures. We plan to work on generating realistic and chemically accepting complex tertiary protein structures. We are also interested in tertiary protein structures as a data augmentation task for specific families of proteins that do not have an adequate amount of protein structures. We are also interested in the conditional GAN as a generative model by employing amino-acid sequences. Finally, we plan to build end-to-end models that start with a tertiary structure and end with different tertiary structures. They have formulated from its under various physiological conditions.

Acknowledgment

I acknowledge the help of the Data Science Lab Department of Information System, Faculty of Computers and Information, Assiut University, Egypt.

References

[1] S. Clayman and J. Heritage, *The news interview: Journalists and public figures on the air*. Cambridge University Press, 2002.

[2] Y. Wang, A. Imran, A. Shami, A. A. Chaudhary, and S. Khan, "Decipher the helicobacter pylori protein targeting in the nucleus of host cell and their implications in gallbladder cancer: An insilico approach," *Journal of Cancer*, vol. 12, no. 23, p. 7214, 2021.

[3] Y. Li, S. Khan, A. A. Chaudhary, H. A. Rudayni, A. Malik, and A. Shami, "Proteome-wide screening for the analysis of protein targeting of chlamydia pneumoniae in endoplasmic reticulum of host cells and their possible implication in lung cancer development," *Biocell*, vol. 46, no. 1, p. 87, 2022.

[4] S. Khan, S. Zaidi, A. S. Alouffi, I. Hassan, A. Imran, and R. A. Khan, "Computational proteome-wide study for the prediction of escherichia coli protein targeting in host cell organelles and their implication in development of colon cancer," *ACS omega*, vol. 5, no. 13, pp. 7254–7261, 2020.

[5] J. Li, M. Zakariah, A. Malik, M. S. Ola, R. Syed, A. A. Chaudhary, and S. Khan, "Analysis of salmonella typhimurium protein-targeting in the nucleus of host cells and the implications in colon cancer: an in-silico approach," *Infection and Drug Resistance*, vol. 13, p. 2433, 2020.

[6] S. Khan, M. Zakariah, C. Rolfo, L. Robrecht, and S. Palaniappan, "Prediction of mycoplasma hominis proteins targeting in mitochondria and cytoplasm of host cells and their implication in prostate cancer etiology," *Oncotarget*, vol. 8, no. 19, p. 30830, 2017.

[7] W. Yu and A. D. MacKerell, "Computer-aided drug design methods," in *Antibiotics*. Springer, 2017, pp. 85–106.

[8] B. Kuhlman and P. Bradley, "Advances in protein structure prediction and design," *Nature Reviews Molecular Cell Biology*, vol. 20, no. 11, pp. 681–697, 2019.

[9] N. Anand and P. Huang, "Generative modeling for protein structures," *Advances in neural information processing systems*, vol. 31, 2018.

[10] T. Rahman, Y. Du, L. Zhao, and A. Shehu, "Generative adversarial learning of protein tertiary structures," *Molecules*, vol. 26, no. 5, p. 1209, 2021.

[11] C. I. Branden and J. Tooze, *Introduction to protein structure*. Garland Science, 2012.

[12] M. Diener, J. Adamcik, A. Sánchez-Ferrer, F. Jaedig, L. Schefer, and R. Mezzenga, "Primary, secondary, tertiary and quaternary structure levels in linear polysaccharides: From random coil, to single helix to supramolecular assembly," *Biomacromolecules*, vol. 20, no. 4, pp. 1731–1739, 2019.

[13] I. Rehman, C. C. Kerndt, and S. Botelho, "Biochemistry, tertiary protein structure," 2017.

[14] S. Hauri, H. Khakzad, L. Happonen, J. Teleman, J. Malmström, and L. Malmström, "Rapid determination of quaternary protein structures in complex biological samples," *Nature Communications*, vol. 10, no. 1, p. 192, 2019.

[15] M. Smyth and J. Martin, "x ray crystallography," *Molecular Pathology*, vol. 53, no. 1, p. 8, 2000.

[16] K. Wüthrich, "The way to nmr structures of proteins," *Nature structural biology*, vol. 8, no. 11, pp. 923–925, 2001.

[17] M. Carroni and H. R. Saibil, "Cryo electron microscopy to determine the structure of macromolecular complexes," *Methods*, vol. 95, pp. 78–85, 2016.

[18] S. C. Pakhrin, B. Shrestha, B. Adhikari, D. B. Kc *et al.*, "Deep learning-based advances in protein structure prediction," *International Journal of Molecular Sciences*, vol. 22, no. 11, p. 5553, 2021.

[19] 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 research*, vol. 28, no. 1, pp. 235–242, 2000.

[20] A, "Uniprot: the universal protein knowledgebase in 2021," *Nucleic acids research*, vol. 49, no. D1, pp. D480–D489, 2021.

[21] A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis, and J. Moult, "Critical assessment of methods of protein structure prediction (casp)—round xiii," *Proteins: Structure, Function, and Bioinformatics*, vol. 87, no. 12, pp. 1011–1020, 2019.

[22] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, no. 7873, pp. 583–589, 2021.

[23] D. D. Boehr, R. Nussinov, and P. E. Wright, "The role of dynamic conformational ensembles in biomolecular recognition," *Nature chemical biology*, vol. 5, no. 11, pp. 789–796, 2009.

[24] S. Majumder, D. Chaudhuri, J. Datta, and K. Giri, "Exploring the intrinsic dynamics of sars-cov-2, sars-cov and mers-cov spike glycoprotein through normal mode analysis using anisotropic network model," *Journal of Molecular Graphics and Modelling*, vol. 102, p. 107778, 2021.

[25] R. Henderson, R. J. Edwards, K. Mansouri, K. Janowska, V. Stalls, S. Gobeil, M. Kopp, D. Li, R. Parks, A. L. Hsu *et al.*, "Controlling the sars-cov-2 spike glycoprotein conformation," *Nature structural & molecular biology*, vol. 27, no. 10, pp. 925–933, 2020.

[26] H. Tian and P. Tao, "Deciphering the protein motion of s1 subunit in sars-cov-2 spike glycoprotein through integrated computational methods," *Journal of Biomolecular Structure and Dynamics*, vol. 39, no. 17, pp. 6705–6712, 2021.

[27] R. Clausen, B. Ma, R. Nussinov, and A. Shehu, "Mapping the conformation space of wildtype and mutant h-ras with a memetic, cellular, and multiscale evolutionary algorithm," *PLoS computational biology*, vol. 11, no. 9, p. e1004470, 2015.

[28] E. Sapin, D. B. Carr, K. A. De Jong, and A. Shehu, "Computing energy landscape maps and structural excursions of proteins," *BMC genomics*, vol. 17, no. 4, pp. 433–456, 2016.

[29] T. Maximova, E. Plaku, and A. Shehu, "Structure-guided protein transition modeling with a probabilistic roadmap algorithm," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 15, no. 6, pp. 1783–1796, 2016.

[30] S. Sabban and M. Markovsky, "Ramanet: Computational de novo protein design using a long short-term memory generative adversarial neural network," *BioRxiv*, p. 671552, 2019.

[31] N. Anand, R. Eguchi, and P.-S. Huang, "Fully differentiable full-atom protein backbone generation," *ACM*, 2019.

[32] H. Yang, M. Wang, Z. Yu, X.-M. Zhao, and A. Li, "Gancon: Protein contact map prediction with deep generative adversarial network," *IEEE Access*, vol. 8, pp. 80 899–80 907, 2020.

[33] W. Ding and H. Gong, "Predicting the real-valued inter-residue distances for proteins," *Advanced Science*, vol. 7, no. 19, p. 2001314, 2020.

[34] M. T. Degiacomi, "Coupling molecular dynamics and deep learning to mine protein conformational space," *Structure*, vol. 27, no. 6, pp. 1034–1040, 2019.

[35] F. F. Alam and A. Shehu, "Variational autoencoders for protein structure prediction," in *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 2020, pp. 1–10.

[36] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[37] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/892c3b1c6dccd52936e27cbd0ff683d6-Paper.pdf

[38] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[39] R. Huang, S. Zhang, T. Li, and R. He, "Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2439–2448.

[40] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/04025959b191f8f9de3f924f0940515f-Paper.pdf

[41] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[42] Y. Ding, C. E. Lawrence, and A. E. Keating, "A deep learning framework for protein structure prediction," *Nature Methods*, vol. 19, no. 2, pp. 131–141, 2022.

[43] S. Ovchinnikov and P. S. Huang, "Achieving high-resolution protein structure prediction with augmented neural networks," *Nature Communications*, vol. 12, no. 1, pp. 1–10, 2021.

[44] J. Pessoa, "Pytorch_msssim: Pytorch implementation of ssim," https://github.com/jorge-pessoa/pytorch-msssim, 2021, accessed: April 8, 2023.

[45] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision*, vol. 40, no. 2, pp. 99–121, 2000.

[46] A. Gretton, K. Borgwardt, M. J. Rasch, B. Scholkopf, and A. J. Smola, "A kernel method for the two-sample problem," *arXiv preprint arXiv:0805.2368*, 2008.

[47] A. Mohammadi and K. N. Plataniotis, "Improper complex-valued bhattacharyya distance," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 5, pp. 1049–1064, 2015.

[48] S. Ma, "Alternating direction method of multipliers for sparse principal component analysis," *Journal of the Operations Research Society of China*, vol. 1, no. 2, pp. 253–274, 2013.

[49] K. You and X. Zhu, *ADMM: An R package for [brief description of the package]*, 2021, r package version 0.3.3. [Online]. Available: https://CRAN.R-project.org/package=ADMM