Advanced Intelligent Discovery

ADVANCED INTELLIGENT DISCOVERY Open Access

RESEARCH ARTICLE OPEN ACCESS

Revealing Protein–Protein Interactions Using a Graph Theory-Augmented Deep Learning Approach

Bahar Dadfar[1] | Gözde Kabay[1] | Matthias Franzreb[1] | Kody Whisnant[2,3] | Nicholas A. Kotov[2,3] | Joerg Lahann[1,2,3]

[1]Karlsruhe Institute of Technology (KIT), Institute of Functional Interfaces (IFG), Eggenstein-Leopoldshafen, Germany | [2]Department of Chemical Engineering, University of Michigan, Ann Arbor, Michigan, USA | [3]Biointerfaces Institute, University of Michigan, Ann Arbor, Michigan, USA

Correspondence: Joerg Lahann (lahann@umich.edu)

Received: 17 November 2025 | **Revised:** 18 December 2025 | **Accepted:** 5 January 2026

Keywords: binding strength classification | feature engineering | graph theory | neural network | protein–protein interaction

ABSTRACT

Understanding protein–protein interactions is crucial in the fields of biology and biotechnology, offering insights into cellular functions and highlighting therapeutic targets. Deep learning (DL) models, particularly convoluted neural networks, have revolutionized biomolecular analysis, yet their success is often tethered to enormous computational costs and reliance on specialized and time-consuming hardware, creating a significant bottleneck for scientific discovery. More specifically, image-based analysis of protein–protein interactions requires time-consuming and computationally intensive quantization of complex structural patterns. In this study, we present an approach for overcoming these computational barriers by leveraging graph theory (GT) and DL approaches synergistically for efficient and accurate classification of protein interactions. Experimentally, this approach utilized multiscale structural features extracted from graphs representing polarized light microscope images of protein stains. Prediction accuracies for categorizing protein binding strengths were as high as 98.67% after feature optimization, including the generation of new augmented descriptors through simple mathematical operations, input dataset size reduction, and feature selection based on minimum redundancy maximum relevance algorithms, while also reducing the training time to only 60 s. Our results highlight the potential for augmenting DL reasoning with GT parametrization, which improves accuracy and reduces training time. These findings establish a powerful and accelerated alternative to conventional image-based DL for analyzing protein–protein interactions. This approach not only provides a cost-effective and accessible alternative to traditional methods, but also facilitates the accurate prediction of protein interaction strengths, even for untrained samples.

1 | Introduction

Proteins serve as the essential components in nearly all biological and biotechnological processes, exhibiting varying degrees of affinity and specificity. Their functionalities are governed by intricate regulatory networks of transient protein–protein interactions (PPIs). Predicting and studying PPIs entail using a diverse array of techniques developed over decades, encompassing in vitro and in vivo assays. Despite exceptional progress, these approaches face limitations such as false positives/false negatives, challenges in obtaining protein crystal structures, and detecting transient PPIs. Experimental methods used to discern interactions also influence their detectability. To circumvent these limitations, novel approaches have gained prominence, aiding in the investigation of PPIs [1–9].

The ongoing reduction in the cost of high-throughput experiments, coupled with advancements in novel computational prediction methods, has led to the generation of extensive PPI datasets. This capability to furnish relatively comprehensive and dependable collections of PPIs has spurred the creation of numerous databases, each with distinct objectives and strengths, aiming to aggregate and consolidate the available data [6]. In our previous studies, we have demonstrated strong correlations

between characteristic changes in their respective deposition stain patterns and trends in the levels of protein–protein interactions [10, 11].

Deep learning (DL)–assisted approaches are well-suited to describe biomolecular interactions [10, 12, 13]. A recent study introduced a DL–assisted approach for classifying the binding affinity of humoral immunoprotein complexes from microscopy images [11]. In this method, droplet assays were prepared by mixing immunoglobulin G (IgG) from various species with a recombinant B-cell superantigen (staphylococcal Protein A) on hydrophobic polymer substrates; the resulting protein stain patterns were captured via polarized light microscopy (PLM) and used as input for a convolutional neural network (CNN) analysis. A pretrained Inception V3 convolution neural network (CNN) architecture was used to analyze ~23,745 such images, enabling both species-specific identification of IgG and inference of their binding affinity levels to Protein A. The DL model achieved an overall classification accuracy of 81.4% across 36 distinct IgG:Protein A binding conditions, with the highest predictive accuracy observed for human IgG (the species exhibiting the strongest known affinity for Protein A). Notably, the network's classification performance was consistent with independent biophysical measurements, where its accuracy trends for different IgG/Protein A mixtures correlated with the binding strengths determined by circular dichroism (CD) spectroscopy, indicating that the model's predictions reflect true protein–protein affinity differences. The trained CNN algorithm also demonstrated robust generalization beyond its training data; when challenged with a novel ligand (Protein G, a different antibody-binding protein not seen during training), it correctly classified human IgG binding patterns with ~94% accuracy across various molar ratios. High-performance outcomes for unfamiliar immunoprotein complexes highlight a model's ability to capture fundamental features of antibody–ligand interactions. Overall, this DL framework illustrates that image-based analysis of simple protein droplet stains can serve as a rapid and accurate proxy for evaluating PPI affinities in the humoral immune system, offering a valuable tool for antibody characterization and informing protein engineering or stability assessments in complex biological media [11].

In recent years, graph theory (GT) has become a powerful tool for analyzing intricate structural data, allowing for the extraction of numerical features that can be conveniently and transparently applied to machine learning-based classification tasks. Instead of directly processing graph structures, researchers have focused on deriving meaningful descriptors from graphs to improve classification accuracy while reducing computational complexity. For instance, studies on brain connectivity networks have leveraged graph-based feature extraction to classify migraine pathology based on MRI images [14]. Similarly, the extraction of GT and other topological parameters from biological sequences such as DNA, RNA, and proteins has been facilitated by computational tools like MathFeature, which applies mathematical descriptors to enhance classification performance [15]. In transcriptomic data analysis, graph-based approaches have been used to optimize feature selection from high-dimensional RNA-Seq datasets, significantly improving classification accuracy by reducing redundant features [16]. In addition, advanced DL techniques, such as graph convolutional network (GCN)-based methods, have been used for selecting biologically relevant features, ensuring a more interpretable and efficient classification process [17]. Zhang et al. demonstrated the potential of GT-based numerical descriptors in enhancing classification performance, highlighting their applicability in improving diagnostic accuracy [18]. These studies collectively highlight the growing potential of graph-theoretic numerical feature extraction in biomedical applications, offering robust solutions for biological classification problems.

In this study, we propose augmentation of DL approaches using the power of GT to quantify the structure of complex structures that would be intractable or exceedingly costly by other means. At the core of our approach is the conversion of complex image data into a more compact, information-dense feature space using graph-based representations. Rather than analyzing raw pixel data with computationally demanding CNNs, we transform the images of multiprotein stains into mathematical graphs. We then extract key numerical descriptors that capture essential topological features of the underlying patterns. In validating this approach, we consider the required time and computational cost of the training process, required for CNN-based image classifications. To overcome the limitations of the CNNs within the training time and computational resources, a novel approach will be adopted involving the conversion of images to graphs. This approach was motivated by the intricate, graph-like structures observed in the captured images from the previous study [11], particularly regarding the changing morphology of salt crystals under various protein mixtures and interactions. In Figure 1A,B, the procedure of obtaining the deposited patterns of Immunoglobulin G (IgG) complexes on the coated surface is shown. The deposited patterns were analyzed using PLM shown in Figure 1C. In Figure 1D, the analysis difference between GT-based neural network and CNN on the images is presented. Given the potential for enhanced feature extraction and simplified analysis offered by graph-based representations, the *StructuralGT* Python package, originally developed at the University of Michigan [19,20], was adapted for this study's specific requirements.

The *StructuralGT* package leverages principles from graph theory to convert images into graphs, enabling the extraction of meaningful features pertinent to the underlying structures depicted in the images [19]. By converting image data of various protein stains into tabulated numerical features through systematic feature engineering and assessment of predictor importance, a compact and informative feature set was identified for training a custom-designed neural network. We then evaluated the performance of GT analysis for optimization of time and computational costs by converting all images into graphs, identifying meaningful features, and training a neural network on the table of features as input dataset.

2 | Experimental Section

2.1 | Dataset Source

The image data already available from our earlier work has been utilized for training, testing, and validation [11]. The dataset comprises a total of 23,745 PLM images, covering 36 distinct interaction conditions of dried protein droplet patterns. These images were captured from IgG complexes of four species (i.e., human, rabbit, bovine, and goat) as well as human serum albumin (HSA) as the negative control and mixed with recombinant

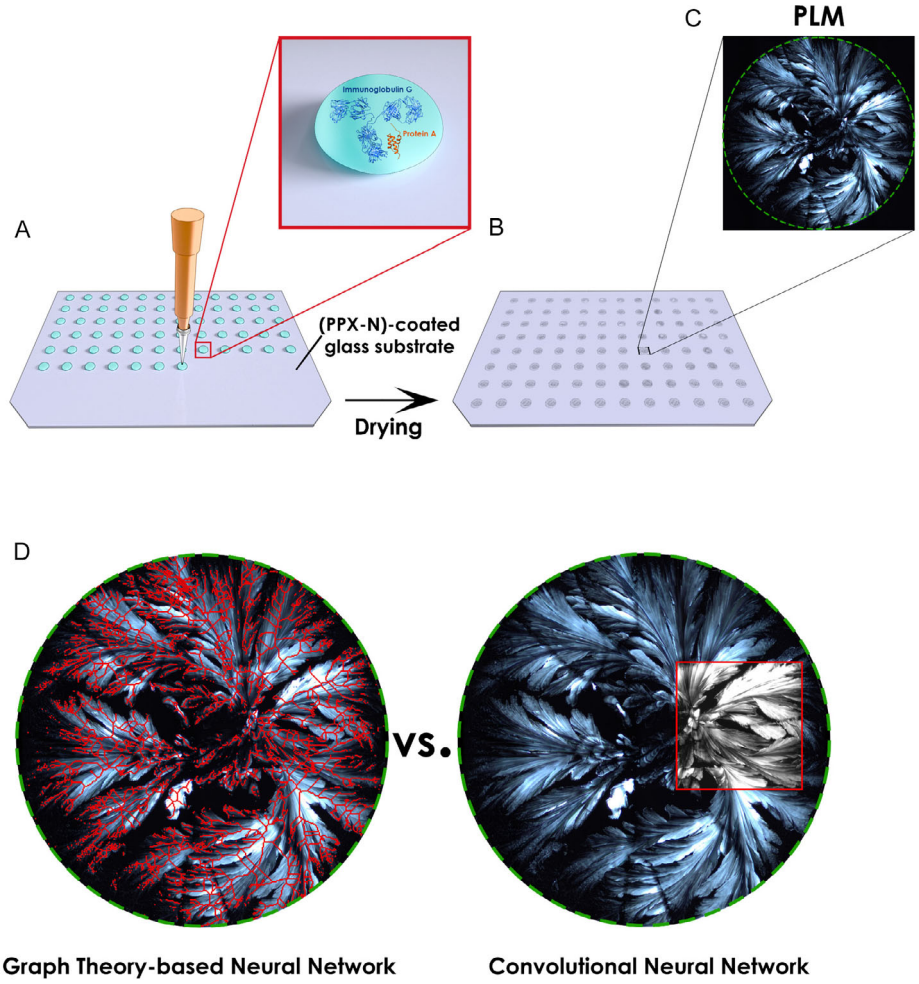

FIGURE 1 | Formation of protein stains and droplet deposition and drying process. (A) Poly(*p*-xylylene)-coated glass substrates and an automated pipetting system were used for dispensing several protein droplets (2 μl) containing different molar ratios of IgG from different species and recombinant Protein A. (B) Dispensed droplets were dried under controlled environmental conditions. (C) PLM imaging was used to collect all the patterns of the deposited proteins. (D) The results of image analysis with two different neural networks were compared.

Protein A (B-cell superantigen (SAg)) at various molar ratios. Each droplet was dispensed onto poly(p-xylylene)-deposited glass substrates, and images of protein stains were obtained using PLM. More details about the sample preparation, PLM imaging, and CNN performance evaluation of our earlier work can be found in the Ref. [11].

2.2 | Graph Features Extraction

Regarding the GT-based analysis of the images, the *StructuralGT* python package was used [19, 20]. Using this package and upon conversion of images to graphs, a comprehensive table of extracted features was generated (Table 1), encompassing 15 meaningful features for each input setting (in this study).

The PLM images primarily report salt crystal morphologies formed during the evaporation of buffered protein solutions. PPIs are therefore not imaged directly. Instead, they influence the physicochemical state of the solution (e.g., degree of complexation, aggregation propensity, and interfacial behavior), which in turn modulates heterogeneous nucleation and crystal growth kinetics during drying [21]. As a result, systematic changes in the morphology of crystal networks can be explained by changes in PPI strength. In the graph representation, increasing interaction strength is associated with larger networks (increasing node and edge counts) but reduced connectivity and ordering (decreasing density, clustering, global efficiency, and closeness). These trends are consistent with a transition from compact, crosslinked crystalline meshes toward more extended, but sparsely connected and fragmented branched structures. Therefore, the GT descriptors serve as quantitative morphology-based proxies for PPI-modulated crystallization behavior, enabling binding-strength classification without claiming direct molecular-scale measurement of affinity. The class-averaged trends of GT descriptors across weak, medium, and strong PPIs are shown in Supporting Information, Figure S1.

In this process, the images were first converted to grayscale and then binarized based on predefined input parameters. Using both the processed grayscale and binary images, final graphical representations were constructed, from which relevant features are extracted. In the image-to-graph conversion process, the selection of input parameters significantly impacted the quality and structure of the resulting binary images. Two key parameters, gamma adjustment (γ) and global threshold value (T),

TABLE 1 | Features extracted from an individual image using GT analysis.

	Unweighted GT parameter	Weighted GT parameter
1	Number of nodes	Weighted average degree
2	Number of edges	Weighted assortativity coefficient
3	Average degree	Width-weighted average betweenness centrality
4	Graph density	Length-weighted average closeness centrality
5	Global efficiency	Width-weighted average eigenvector centrality
6	Average clustering coefficient	—
7	Assortativity coefficient	—
8	Average betweenness centrality	—
9	Average closeness centrality	—
10	Average eigenvector centrality	—

All these features are further detailed in Table S1.

played a crucial role in determining the characteristics of these binary images.

Gamma correction (γ) is a nonlinear transformation applied to grayscale images to adjust brightness and contrast. A lower gamma value ($\gamma < 1$) enhanced darker regions, while a higher gamma value ($\gamma > 1$) suppressed brighter regions [22]. Global thresholding (T) is a binarization technique in which a fixed intensity threshold is applied across the entire grayscale image, distinguishing foreground from background [22, 23].

The image-to-graph conversion relies on gamma correction (γ) and global thresholding (T) to extract structural information from PLM images at different levels of detail. Lower gamma values, combined with higher global thresholds, emphasized the most prominent structural features, yielding graph representations that highlight dominant aspects of the patterns. Conversely, increasing γ while reducing T allowed for finer details to be incorporated into the resulting graphs, preserving intricate features within the patterns. Given the complexity of the macroscopic to the microscopic aspects of the observed crystalline patterns, rather than optimizing a single γ–T pair, five distinct parameter settings, γ values (0.9, 1.0, 1.3, 1.5, and 2.5), corresponding T values (140, 127, 100, 70, and 80), were systematically optimized to ensure the most detailed and representative graph extraction from the images as shown in Figure 2. Importantly, the final classification performance does not depend on a specific γ–T setting. GT descriptors are extracted across all parameter combinations and aggregated prior to feature selection (explained in the following sections), enabling a multiscale representation of the salt crystal networks. This approach reduces sensitivity to preprocessing choices and mitigates overfitting to a particular binarization condition.

By systematically varying these input parameters across the five selected settings, a comprehensive range of binary images capturing different levels of detail and complexity within the patterns were generated. This approach ensured that the resulting set of features derived from the graph representations encompassed a broad spectrum of information, from the macroscopic to the microscopic aspects of the observed patterns. All the images used in our previous study were used for this purpose. These images were the patterns of dried droplets of various immunoglobulins mixed with a B-cell superantigen (SAg) (recombinant Protein A), which were deposited onto hydrophobic polymer substrates and the resulting protein stains were imaged using PLM. A comprehensive study based on 23,745 images was carried out in the previous work [11].

2.3 | Augmented Features

To further enhance the neural network's classification performance, additional features were derived from the original GT descriptors (Table 1) through simple mathematical operations. Specifically, summation and multiplication were applied to existing feature values to augment the dataset with new information that could reveal subtle patterns and relationships. These operations were chosen for their simplicity and effectiveness in capturing feature interactions without introducing excessive complexity. Summation was used to represent cumulative relationships, effectively capturing overall structural characteristics of the protein complexes, while multiplication was used to model interactions and correlations between features [24]. The augmented features were constructed as simple algebraic combinations of physically interpretable GT metrics to capture joint morphology effects without introducing opaque transformations. For example, (number of nodes (n) + number of edges (e)) summarizes overall crystalline skeleton extent and segmentation, reflecting how much branched structure is present in the salt crystal network. Multiplying this quantity by the Average Clustering Coefficient (ACC), $\Delta \times (n + e)$, yields an "ordered extent" index that distinguishes extended but sparsely connected dendritic growth from large, locally crosslinked mesh-like crystallization. This systematic feature augmentation enriched the dataset and contributed to the improved accuracy of protein–protein interaction classification without increasing computational costs. By integrating these derived features with original graph-theoretic ones, the network gains a more comprehensive representation, improving its ability to discern patterns and make accurate predictions. Accordingly, the feature "average clustering coefficient (ACC)" was selected. ACC in graph theory measures the fraction of all possible triangles (three-node subgraphs) in a graph, averaged over all nodes. In protein pattern analysis, this feature is crucial as it highlights structural variations between different protein stains, providing a quantifiable measure of

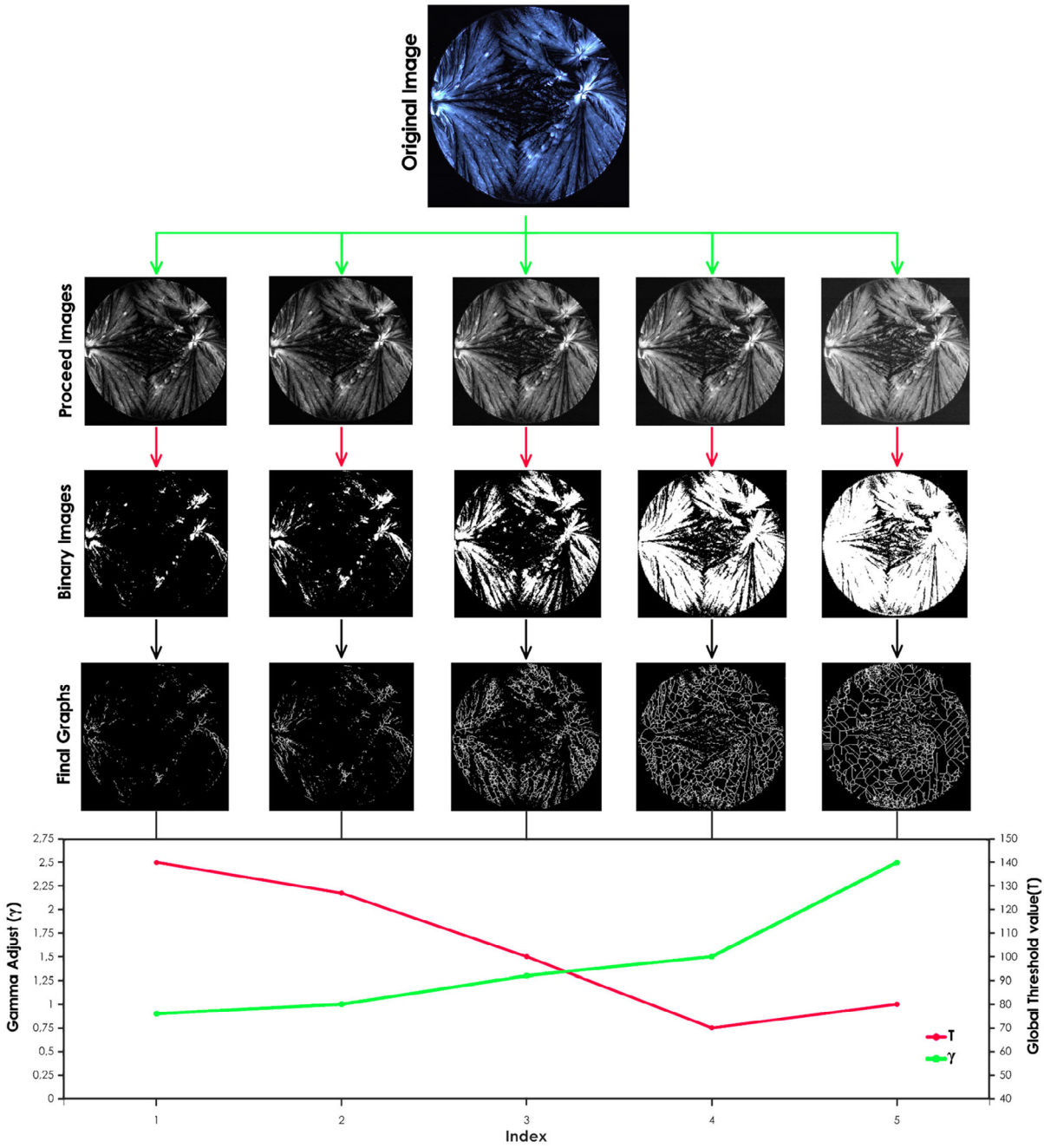

FIGURE 2 | Image-to-graph conversion for GT analysis. Example of an image dataset for modifying and applying five different gamma adjustments and global threshold values as input parameters to capture intricate details of the given images. Starting from the original PLM image (top), five different parameter combinations of γ-T are applied. Each column shows the sequential processing steps: From top to bottom: an original PLM image, processed grayscale images after γ correction, binary images after global thresholding, and final graph representations. The bottom panel illustrates the inverse relationship between γ adjustment values (green line, left y-axis) and global threshold values (red line, right y-axis) across the five (γ-T) parameter settings demonstrated in indices 1–5, showing the systematic variation used to capture different levels of structural detail within the crystalline patterns.

the overall connectivity and efficiency of the graph. This directly correlates with the structural integrity and interaction patterns of proteins. Motivated by this fact, we further explored its potential by incorporating it into the mathematical operations described previously, namely, summation and multiplication, to derive additional aggregated features. One approach involved multiplying ACC by the sum of weighted and nonweighted features, creating composite metrics that encapsulate more structural information. This method combines the efficiency of the graph with specific structural attributes, providing a more robust indicator for classification. Another derived feature involved summing the number of nodes and the number of edges within graphs. This aggregated feature can reveal more about the graph's complexity and density than considering nodes and edges separately. By combining the number of nodes and edges, the feature captures both the size and connectivity of the graph, offering a more comprehensive view of its structure. These augmented features are described in Table 2.

TABLE 2 | Augmented features.

New feature	Formula (n = # of nodes, e = # of edges, subscript w refers to the weighted feature)	Additional description
Sum nodes edges	$\Delta \times (n + e)$	Δ is the average clustering coefficient, the fraction of neighbors of a node that are directly connected to each other as well (forming a triangle)
Sum assortativity	$\Delta \times (r + r_w)$	The assortativity coefficient, r, measures similarity of connections by node degree
Sum degree	$\Delta \times (k + k_w)$	k, the degree of a node is the number of edges connected to that particular node
Sum betweenness centrality	$\Delta \times (CB + CB_w)$	CB, the betweenness centrality is defined as the number of shortest paths that pass through a particular node in a graph
Sum closeness centrality	$\Delta \times (CC + CC_w)$	CC, the closeness centrality is the reciprocal of the average shortest distance from a node to all other nodes in the graph
Sum eigen vector centrality	$\Delta \times (x + x_w)$	Eigenvector centrality (x) is a measure of the influence of a node in a graph.

To assess the contribution of original as well as the new calculated features to the classification task and to identify the most informative descriptors for subsequent optimization, a feature selection process was implemented to all extracted and augmented feature set. Specifically, the minimum redundancy maximum relevance (MRMR) algorithm was used to evaluate the importance of each input feature based on its relevance to the target classes (36 labels) and redundancy with other features. This analysis provided a systematic approach for screening and prioritizing features that contribute most significantly to the predictive performance of the model. The MRMR feature selection analysis further supports the physical interpretability of the proposed framework. As shown in Figure 3, the highest-ranked predictors are dominated by GT descriptors related to network size (number of nodes and edges), connectivity and sparsity (graph density, average degree), local ordering (average clustering coefficient), and global transport properties (global efficiency, closeness, and betweenness centrality). Notably, the

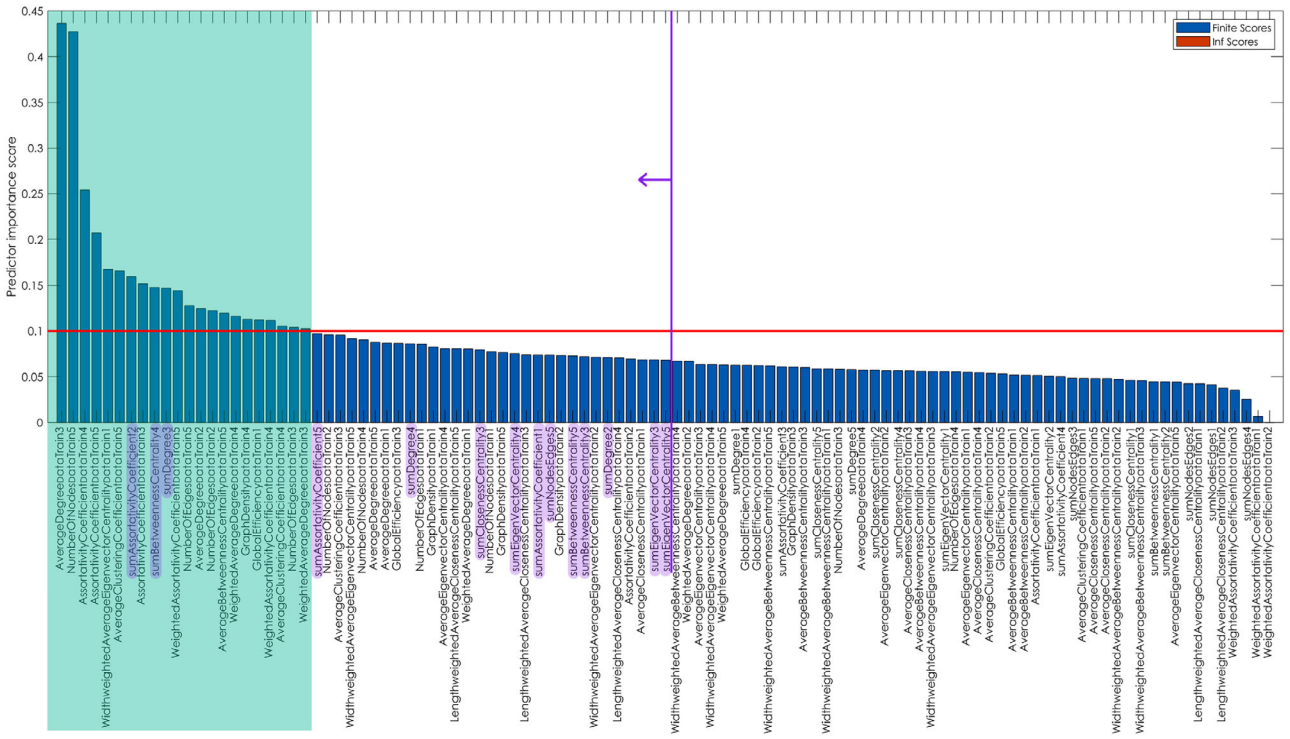

FIGURE 3 | Feature selection scores. Scores for the extracted/computed features using an MRMR feature selection algorithm. The green box shows the selected features with a threshold score of 0.1 or higher. The purple line indicated the 50th percentile in the ranking relative to the entire input features which includes six augmented features (highlighted in purple), demonstrating their relevance for classification tasks.

augmented composite descriptors are also retained, with all exceeding the 50th percentile (left side of the purple line) in the feature importance ranking relative to the entire input feature set as shown in Figure 3, indicating that joint information on network extent, local crosslinking, and geometric prominence of crystal branches is critical for discrimination of interaction strength. These features are preferred because they exploit systematic morphology changes in salt crystal networks modulated by protein–protein interactions, instead of arbitrary or purely statistical features. Therefore, these features were used for the following classification tasks. The number at the end of the feature names in Figure 3 is the index related to the input parameter used for graph construction described previously.

2.4 | Neural Network Design

The MATLAB programming software (Release 2023a, MathWorks Inc.) was used to develop a neural network tailored for training and classifying features extracted from graphs. Given that the input dataset comprised a table consisting of numerical values representing each label, a straightforward neural network architecture with 16 layers was deemed suitable for the classification task.

2.5 | Training and Testing of Neural Network

The designed neural network was configured with specific parameters to ensure efficient and effective training. The training procedure involved several key steps.

- Neural Network Configuration: The neural network was configured with a mini-batch size of 128 and a maximum number of epochs set to 80. The ADAM optimizer was used as the solver algorithm for optimizing the network's weights and biases.
- Activation Function: The activation function used in the hidden layers of the neural network was the Exponential Linear Unit (ELU). This choice was based on empirical evidence suggesting that ELU outperformed Rectified Linear Unit in terms of accuracy during training.
- Validation Set: To assess the performance of the trained model and prevent overfitting, 10% of the input dataset was randomly selected to serve as the validation set. This validation set was disjoint from the training dataset to ensure unbiased evaluation.
- Testing dataset: For evaluating the performance of the trained model, the same distinct converted images used in image classification were used. This consistency ensured an accurate comparison of the model's performance across different tasks.
- Feature Selection: The MRMR algorithm was applied to identify the most effective features for training. Each feature was assigned an importance score, and features with scores below the threshold of 0.1 were removed from the dataset to reduce redundancy and noise.

In addition, to expedite the feature extraction process and minimize computational overhead, a reduced input dataset size was created by randomly selecting 10% of the initial input dataset. The training procedure was then repeated using the same architecture on this reduced dataset.

3 | Results and Discussion

The optimization strategy aimed to minimize training time and computational cost while maintaining high classification accuracy. To achieve this, we leveraged graph theory, a robust mathematical framework for analyzing complex structures [19]. The graph-based representation is not designed to mimic the visual feature extraction performed by CNNs. Instead, it provides a mathematical abstraction of the connectivity and organization of salt crystal networks formed during evaporation, enabling quantification of morphology features such as branching complexity, fragmentation, and dominance of growth pathways in different PPI strengths that are difficult to isolate directly from raw pixel data. Instead of directly utilizing the graphs, we focused on deriving pertinent numerical attributes that effectively encapsulate their topology described previously. These extracted features formed the input dataset for the classification model.

3.1 | Neural Network Architecture

A custom-designed neural network was developed to train and classify the extracted graph features efficiently. The architecture commenced with a feature input layer tasked with ingesting the tabular data containing 17,345 rows and 105 columns ([number of graph theory defined meaningful features (15) + number of new augmented composite features (6)] × number of input parameters (5)), representing the extracted features. Subsequently, a batch normalization layer was introduced to normalize the input data, followed by an activation layer to introduce nonlinearity and facilitate feature transformation. A pivotal early fully connected layer was strategically incorporated to aid in the categorization process based on the extracted features. This layer, with a size of 10 times the number of classes, played a crucial role in enabling the network to discern patterns and make informed classification decisions. To refine the classification process and mitigate the influence of unwanted weights, a tandem of batch normalization and activation layers was used. These layers worked synergistically to cleanse the network of any extraneous information that could hinder accurate classification. A dropout layer was subsequently introduced to prevent overfitting and enhance the network's generalization capabilities. Following this, a second fully connected layer, sized according to the number of classes, was integrated into the architecture to further refine the classification process. The architecture culminated with a softmax layer, responsible for converting the network's raw output into probability scores corresponding to each class, and a classification layer for assigning the final class label. Notably, this architecture diverges from conventional approaches by incorporating an early fully connected layer, which proved instrumental in facilitating effective training and classification. Comparative analyses with other classification techniques, including support vector machine (SVM), decision trees, k-nearest neighbor (KNN), and simple regression models, underscored the superior accuracy achieved by the designed neural network [14, 25–28].

3.2 | Classification of IgG:Protein A Complexes

Following the development of a custom-designed neural network, the model was used for the classification of extracted

features derived from protein patterns. Each feature set was labeled consistently with the image classification task performed using InceptionV3 CNN to maintain coherence in the classification frameworks [11]. Adhering to the same methodological pipeline for image classification, the network underwent training and testing using a designated test dataset comprising 3600 image data sets to assess its performance and generalization capability.

Figure 4 illustrates the resulting confusion matrix obtained from the feature classification process by the custom-designed neural network. Notably, the total accuracy achieved was 58.25%, which marked a decrease compared to the accuracy attained in image classification (81.42%) in our previous study performed by the InceptionV3 CNN [11]. The decline in accuracy can be attributed to the network's tendency to identify similarities in the strength of interactions, particularly evident in cases such as rabbit IgG: Protein A and bovine IgG:Protein A. Further analysis revealed that rabbit IgG and bovine IgG both exhibited medium strength of interactions with Protein A, as determined by CD spectroscopy measurements [11]. While image classification excelled in identifying the source of IgG, it struggled to discern differences in the strength of interaction between different IgG species. Conversely, feature classification enabled more nuanced monitoring of these similarities in interaction strength.

A transparent yellow box under the magnifier in the confusion matrix highlights misclassifications stemming from similarities in interaction strength between IgG from rabbit and bovine serum and Protein A. In addition, misclassifications were more prevalent in different molar ratios within each IgG species class, again reflecting similarities in interaction strength confirmed by CD spectroscopy analysis in our previous study [11].

The local accuracies of each IgG species image class followed a consistent trend, with higher accuracies associated with stronger interaction strengths. Specifically, accuracies of 81.86%, 56.28%, 54.14%, 50.43%, and 44.86% were achieved for IgG from human, rabbit, bovine, goat, and HSA, respectively, in interaction with recombinant Protein A.

Training the feature classification neural network required, ~7 minutes. The network excelled in recognizing interaction strength over molar ratios or protein sources. To leverage this capability, three main labels were chosen based on relative interaction strength: "weak" (goat IgG:Protein A and HSA:Protein A), "medium" (rabbit IgG:Protein A and bovine IgG:Protein A), and "strong" (human IgG:Protein A). The single-protein images were excluded from this revised input dataset because they represent individual proteins without binding partners, making them irrelevant for PPI classification tasks that require evaluation of binding strength between two or more proteins. Upon relabeling the features and retraining the network with the same parameters, a confusion matrix was obtained, as depicted in Supporting Information, Figure S2. Impressively, a total accuracy of 99.83% was achieved in 6 min of training. These results underscore the high performance of using extracted features derived from GT algorithms in classifying PPI strengths.

A key bottleneck was the feature extraction process, which required significant time despite being less than that of the image classification procedure. To optimize the designed neural network, efforts focused on reducing the time required for initial feature extraction and identifying the most effective features for training.

3.3 | Input Dataset Size Reduction

Given the high accuracy achieved in the previous feature classification (99.83%) shown in Figure S2, a subset comprising 10% of

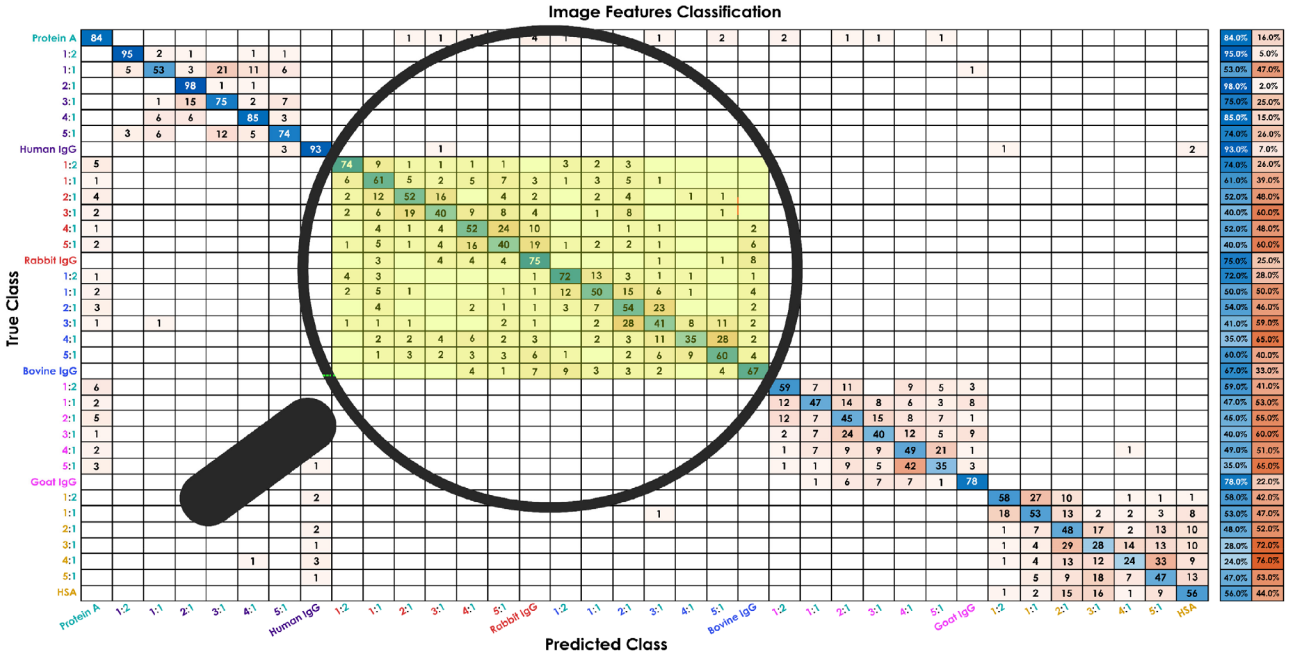

FIGURE 4 | Confusion matrix of different protein mixtures using a feature classification neural network. The matrix displays predicted class (*x*-axis) versus true class (*y*-axis) for 36 interaction conditions, with numbers indicating sample counts. The additional columns on the right side show the row-wise classification accuracy and corresponding error rate for each true class, enabling direct comparison of per-class performance. The diagonal blue boxes indicate correct classifications, while off-diagonal elements in orange shades denote misclassifications between different interaction categories. Each class contains 100 test samples in this study. The total accuracy of this confusion chart is 58.25%.

the input labels was randomly selected and separated for training. Consequently, instead of working with a table of 13,806 rows (excluding single protein classes) and 105 columns, a reduced dataset of 1380 rows and 105 columns was introduced to the neural network as the input dataset. This reduction aimed to address potential training inefficiencies arising from unnecessary data and to streamline the feature extraction process. This input size reduction was also applied to the test dataset.

In this revised approach, random patterns of different protein mixtures were utilized for training, without consideration for protein source or molar ratio, as long as the total mass concentration remained constant. This approach aimed to generalize the network's learning beyond specific protein sources or ratios, facilitating broader applicability.

The resulting confusion matrix, depicted in Figure 5B, achieved a total accuracy of 99.33% with a reduced training time of only about 1 min. This significant improvement in both the required time for feature extraction and training, while maintaining a high accuracy of prediction, underscores the effectiveness of the optimization strategy.

Since the input dataset size was reduced for feature classification, a similar approach was applied to image classification to enable an appropriate comparison between the two methods. The same procedure involved relabeling the image classes and randomly reducing the input image dataset (both training and testing) to 10% of the original size. InceptionV3, a pretrained CNN, was utilized for this reduced image classification task. The training process for this reduced dataset took, approximately, 8 h, resulting in a total prediction accuracy of 99.00% (Figure 5A).

By implementing these changes, we can make a direct comparison between feature classification and image classification in terms of training time and accuracy. The reduction in dataset size significantly improved the efficiency of the feature classification process, achieving similar high accuracy in a fraction of the time required for image classification. Moreover, processing a table of numerical values is not only easier than processing images, but also allows classification to be performed on simple, non-GPU-dependent computing systems, reducing the required training time.

3.4 | MRMR Feature Selection Algorithm

Further optimization can involve selecting the most effective features that contribute to the final decision, as mentioned in the Experimental Section. Several feature selection algorithms are available, among which MRMR is considered highly effective. MRMR assigns each input feature a predictor importance score, indicating its effectiveness in the classification process. In this

FIGURE 5 | Confusion matrix of a reduced-size input dataset for PPI strengths. Randomly, 10% of the input data were selected. (A) Image classification using InceptionV3, a pretrained CNN with prediction accuracy of 99.00%. (B) Feature classification using the designed neural network with prediction accuracy of 99.33%. Both approaches demonstrate excellent performance with minimal misclassifications, primarily occurring between medium and weak interaction classes.

study, features with an importance score threshold of 0.1 or higher were selected, while the others were removed before being introduced to the neural network. Consequently, 22 out of 105 features were retained. The names of the selected features (highlighted in the green transparent box) and their importance scores are shown in Figure 3. Consistent with this design, the MRMR feature selection algorithm consistently retained descriptors related to network size, connectivity, ordering, and pathway dominance across γ–T variations. In Figure 3, the numerical suffix appended to each feature name denotes the corresponding γ–T parameter combination (Index), used for image-to-graph conversion, with each instance treated as a distinct candidate feature during selection. This indicates that the selected features capture robust, intrinsic morphology characteristics of the crystallization patterns rather than artifacts arising from a specific preprocessing configuration or relying on opaque or arbitrary descriptors. As a result, the proposed pipeline remains stable under moderate perturbations of γ and T, providing practical flexibility for application to different imaging setups and stain intensities. The training was conducted using the reduced feature set (input dataset size reduced previously), and the prediction on the test dataset followed. The obtained confusion matrix is shown in Figure 6A, indicating a total prediction accuracy of 98.67% and a training time of about 1 min. These results confirm that feature selection algorithms can enhance the performance of feature classification tasks. Algorithms like MRMR are preferable due to their efficiency in memory consumption, required time, performance, and the explainability of the results [29–32].

3.5 | Feature Classification Generalization

To evaluate the performance of the pretrained neural network (reduced-size dataset with the most effective features) in predicting unknown samples, the images of human IgG:Protein G complexes were converted to graphs, and the necessary 22 features determined by the MRMR algorithm were extracted. The same procedure used for test dataset preparation was followed. Given that human IgG exhibits a strong interaction with Protein G, all the images were labeled as "Strong" and introduced to the trained network to observe its classification accuracy for these PPI classes.

The resulting confusion chart for the unknown sample prediction is presented in Figure 6B. The trained neural network accurately categorized 90.07% of the human IgG:Protein G images as exhibiting a "Strong" strength of interaction. This classification is consistent with CD spectroscopy measurements done in our previous study [11]. These results highlight the neural network's effectiveness in predicting the interaction strength of previously unseen samples based on features extracted from biological graph-like images that are typical for many molecules and tissues, confirming its generalization capability. This successful classification of the human IgG:Protein G system illustrates a general deployment protocol for applying trained GT-augmented models to new protein–ligand combinations. As summarized in Figure 7, PLM images are first acquired from dried protein mixtures and converted into graph representations using the *StructuralGT* python package based on the established γ–T parameter set. GT descriptors are then extracted, augmented through simple algebraic combinations to generate composite features, and reduced to a compact, informative feature set via MRMR-based selection. Finally, the selected features are provided as input to the pretrained neural network to predict interaction strength. Because the model operates on physically interpretable structural descriptors rather than raw pixel data, it can be applied to previously unseen protein mixtures without retraining, provided that the underlying crystallization and pattern formation mechanisms remain comparable.

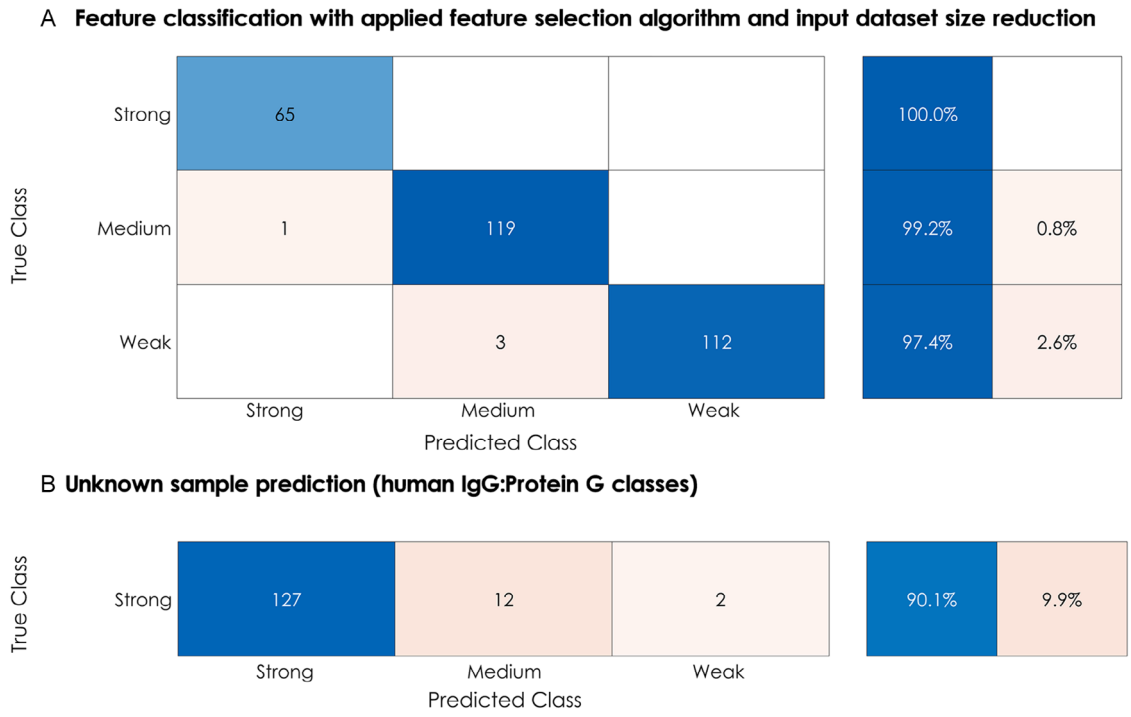

FIGURE 6 | (A) Confusion matrix for PPI strength using feature a classification with input size reduction and an MRMR feature selection algorithm, featuring a prediction accuracy of 98.67%. (B) Confusion matrix of human IgG:Protein G class prediction of unknown samples.

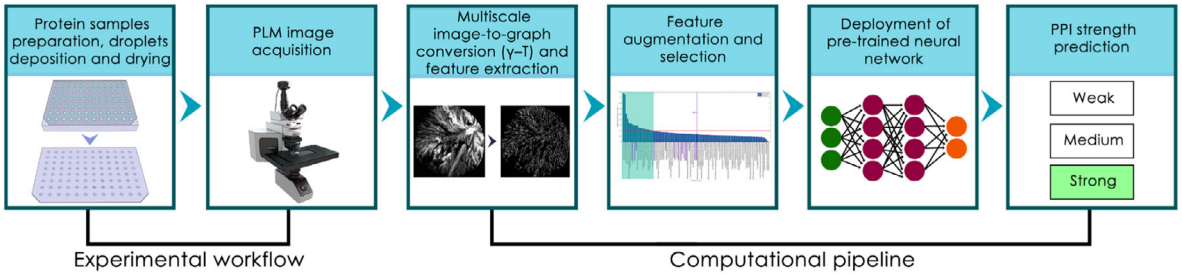

FIGURE 7 | Deployment workflow for GT-augmented PPI strength classification. Following GT feature extraction, augmented composite features are generated using simple algebraic combinations and selected using the MRMR algorithm. The selected feature set is then used by the pretrained neural network to predict interaction strength.

For a comprehensive comparison between image classification and feature classification, Table 3 summarizes the key parameters affecting the final performance of each classification method. This table outlines the required training time and the computational costs associated with each method, both in their original schemes and after the optimizations described in previous sections.

We specifically evaluated the training time, accuracy, and computational cost for both the original and the optimized scheme. The original training time for image classification was significantly higher than feature classification. Image classification required ~29 351 min (about 20 days) compared to 11,570 min (about 8 days) for feature classification for 5 different input parameters for graph theory analysis. After optimization, the training time for image classification is reduced to about 480 min (8 h), while feature classification training time is reduced to just 1 min. This demonstrates a threefold reduction in total required time for feature classification compared to image classification. Meanwhile, the accuracy of the original image classification method (81.42%) is higher than that of the original feature classification method (58.25%). After optimization, both methods achieved high accuracy levels, with image classification reaching 99.00% and feature classification achieving 98.67%. Importantly, image classification is computationally intensive and heavily dependent on GPU resources, whereas feature classification significantly reduced computational costs and did not depend on GPUs, making it more accessible and feasible for various applications. Both methods benefited from reducing the training data size to 10% of the original dataset, maintaining a high accuracy with a smaller dataset, which highlights the efficiency of the optimization process. Another crucial aspect is that feature classification maintained high performance without relying on GPU resources, ensuring consistent results across various computing environments. Image-based classification was performed using a GPU-accelerated CNN running on an NVIDIA TITAN RTX (24 GB) with an Intel Xeon W-2123 CPU (4 cores), whereas GT-based feature classification was executed on a CPU-only system using an AMD Ryzen 7 processor (8 cores) with integrated Radeon Graphics. This comparison highlights that the GT-based pipeline achieves comparable or improved performance without reliance on dedicated GPU acceleration, while CNN-based image classification requires substantially more specialized hardware. The two approaches, therefore, differ fundamentally in computational scaling. CNN-based pipelines scale steeply with dataset size, image resolution, and network depth, whereas the GT-based pipeline shifts most computational cost to a one-time feature extraction step followed by efficient training on a compact tabular dataset.

TABLE 3 | Comparison of image classification and feature classification.

Performance parameter	Image classification	Feature classification
Input dataset preparation time (original)	2400 min (~2 days)	11 563 min (~8 days)
Input dataset preparation time (optimized)	240 min	289 min
Training time (original)	26 951 min (~18 days)	7 min
Training time (optimized)	480 min (8 h)	1 min
Total required time (original)	29 351 min (~20 days)	11 570 min (~8 days)
Total required time (optimized)	720 min (12 h)	290 min (~5 h)
Accuracy (original)	81.42%	58.25%
Accuracy (optimized)	99.00%	98.67%
Computational cost	High (GPU dependent)	Low (Non-GPU dependent)
GPU model	NVIDIA TITAN RTX 24GB	Radeon Graphics
CPU model/number of cores	Intel Xeon W-2123/4	AMD Ryzen 7/8
Training data size reduction	Yes (10% of original size)	Yes (10% of original size)
Performance consistency	High with GPU	High without GPU

4 | Conclusions

Feature classification emerges as a highly efficient and practical alternative to image classification, particularly when computational resources and training time are critical constraints. The significant reduction in training time, which was 2.5 times faster than the optimized CNN, and computational cost, combined with high prediction accuracy as presented in Table 3, underscores the potential of feature classification for broader applications in protein interaction analysis and other biological classification tasks. This optimized approach offers a reliable, cost-effective, and accessible solution for the classification and screening of biological images, facilitating advancements in proteomics research and related fields. This method, implemented using the *StructuralGT* python package, transforms images into graphs and extracts meaningful features for classification.

Overall, these findings taken together suggest that GT-augmented neural networks present a computationally efficient alternative to traditional image-based DL methods for pattern analysis, achieving significant improvements in training and processing speed while maintaining classification accuracy and making advanced pattern recognition tools more accessible in resource-limited settings.

Future work should focus on extended optimization and application of this technique, potentially enhancing its utility in various scientific fields such as proteomics, bioinformatics, and systems biology. The integration of GT with DL represents a powerful toolset for advancing our understanding and capability in protein interaction analysis and beyond, for example, interactions of nanoparticles and proteins as well as nanoparticles and nanoparticles; with both exhibiting direct similarities with PPI. [33, 34].

Acknowledgments

This research was supported by the Helmholtz Foundation program "Materials System Engineering" (MSE) at the Karlsruhe Institute of Technology. Parts of the study were supported by financial support from the National Science Foundation (NSF) and specifically from Grant No. 2243104, Center for Complex Particle Systems (COMPASS).

Funding

This study was supported by the Helmholtz (Materials System Engineering) and NSF (#2243104).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

All image-to-graph conversion steps were implemented using the *StructuralGT* Python package with fixed, and documented parameter settings. Feature extraction, augmentation, and MRMR-based selection follow deterministic procedures that can be readily reproduced. The extracted feature tables, and parameter configurations will be made available upon reasonable request, with plans for public release in a dedicated repository to facilitate reuse and extension of the proposed framework.

References

1. W. Cai and H. Hong, *Protein-Protein Interactions: Computational and Experimental Tools* (BoD–Books On Demand, 2012).

2. J. M. Howell, T. L. Winstone, J. R. Coorssen, and R. J. Turner, "An Evaluation of In Vitro Protein–Protein Interaction Techniques: Assessing Contaminating Background Proteins," *Proteomics* 6 (2006): 2050.

3. H. Watanabe, H. Matsumaru, A. Ooishi, et al., "Optimizing pH Response of Affinity between Protein G and IgG Fc," *Journal of Biological Chemistry* 284 (2009): 12373.

4. Z. Nikolovska-Coleska, *Protein-Protein Interactions: Methods and Applications* (Springer, 2015), 109.

5. A. Velazquez-Campoy, S. A. Leavitt, and E. Freire, *Protein-Protein Interactions: Methods and Applications* (Springer,2015), 183.

6. D. Szklarczyk and L. J. Jensen, *Protein-Protein Interactions: Methods and Applications* (Springer, 2015), 39.

7. I. A. Taylor, K. Rittinger, and J. F. Eccleston, *Protein-Protein Interactions: Methods and Applications* (Springer, 2015), 205.

8. W. E. Stites, "Protein-Protein Interactions: Interface Structure, Binding Thermodynamics, and Mutational Analysis," *Chemical Reviews* 97 (1997): 1233.

9. T. Ehrenberger, L. C. Cantley, and M. B. Yaffe, *Protein-Protein Interactions: Methods and Applications* (Springer, 2015), 57.

10. A. Jeihanipour and J. Lahann, "Deep-Learning-Assisted Stratification of Amyloid Beta Mutants Using Drying Droplet Patterns," *Advanced Materials* 34 (2022): 2110404.

11. B. Dadfar, S. Vaez, C. Haret, et al., "Deep-Learning-Assisted Affinity Classification for Humoral Immunoprotein Complexes," *Small Structures* 5 (2024): 2400204.

12. S. Vaez, B. Dadfar, M. Koenig, M. Franzreb, and J. Lahann, "Deep Learning-Based Classification of Histone–DNA Interactions Using Drying Droplet Patterns," *Small Science* 4 (2024): 2400252.

13. S. Vaez, D. Shahbazi, M. Koenig, M. Franzreb, and J. Lahann, "Deep Learning Based Surface Classification of Functionalized Polymer Coatings," *Langmuir* 41 (2025): 11272.

14. F. Jorge-Hernandez, Y. Garcia Chimeno, B. Garcia-Zapirain, A. Cabrera Zubizarreta, M. A. Gomez Beldarrain, and B. Fernandez-Ruanova, "Graph Theory for Feature Extraction and Classification: A Migraine Pathology Case Study," *Bio-Medical Materials and Engineering* 24 (2014): 2979.

15. R. P. Bonidia, D. S. Domingues, D. S. Sanches, and A. C. de Carvalho, "MathFeature: Feature Extraction Package for DNA, RNA and Protein Sequences Based on Mathematical Descrip," *Briefings in Bioinformatics* 23 (2022): bbab434.

16. C. Gakii, P. O. Mireji, and R. Rimiru, "Graph Based Feature Selection for Reduction of Dimensionality in Next-Generation RNA Sequencing Datasets," *Algorithms* 15 (2022): 21.

17. C. Chen, S. T. Weiss, and Y.-Y. Liu, "Graph Convolutional Network-Based Feature Selection for High-Dimensional and Low-Sample Size Data," *Bioinformatics* 39 (2023): btad135.

18. H. Zhang, C. Li, S. Ai, et al., "Application of Graph-Based Features in Computer-Aided Diagnosis for Histopathological Image Classification of Gastric Cancer," *Digital Medicine* 8 (2022): 15.

19. A. Kadar, W. Wu, A. Emre, S. Glotzer, and N. Kotov, "Graph Representations of Nanostructured Hedgehog Particles with Variable Complexity," *Bulletin of the American Physical Society* (2024).

20. D. A. Vecchio, S. H. Mahler, M. D. Hammig, and N. A. Kotov, "Structural Analysis of Nanoscale Network Materials Using Graph Theory," *ACS Nano* 15 (2021): 12847.

21. T. Yakhno, "Salt-Induced Protein Phase Transitions in Drying Drops," *Journal of Colloid and Interface Science* 318 (2008): 225.

22. R. C. Gonzalez, *Digital Image Processing* (Pearson Education India, 2009).

23. N. Otsu, "A Threshold Selection Method from Gray-level Histograms," *Automatica* 11 (1975): 23.

24. K. Xu, W. Hu, J. Leskovec, and S. Jegelka, arXiv preprint arXiv:1810.00826 (2018).

25. Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Trans Pattern Anal Mach Intell*, 2013, pp. 1798, 35.

26. I. H. Witten, E. Frank, M. A. Hall, C. J. Pal, and M. Data, *In Data Mining*. Elsevier Amsterdam, The Netherlands, 2005). 403–413.

27. W. Rawat and Z. Wang, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review," *Neural Computation* 29 (2017): 2352.

28. Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature* 521 (2015): 436.

29. Z. Zhao, R. Anand, and M. Wang, in *2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)* (IEEE, 2019), 442–452.

30. J. R. Berrendero, A. Cuevas, and J. L. Torrecilla, "The mRMR Variable Selection Method: A Comparative Study for Functional Data," *Journal of Statistical Computation and Simulation* 86 (2016): 891.

31. J. Hermo, V. Bolón-Canedo, and S. Ladra, "Fed-mRMR: A Lossless Federated Feature Selection Method," *Information Sciences (New York)* 669 (2024): 120609.

32. S. Xie, Y. Zhang, D. Lv, X. Chen, J. Lu, and J. Liu, "A New Improved Maximal Relevance and Minimal Redundancy Method Based on Feature Subset," *The Journal of Supercomputing* 79 (2023): 3157.

33. M. Cha, E. S. T. Emre, X. Xiao, J.-Y. Kim, et al., "Unifying Structural Descriptors for Biological and Bioinspired Nanoscale Complexes,"*Nature Computational Science* 2 (2022):243

34. F. C. Meldrum, N. A. Kotov, J. H. Fendler, "Monoparticulate Layers of Titanium Dioxide Nanocrystallites with Controllable Interparticle Distances,"*Journal of Physical Chemistry* 98 (1994); 4506.

Supporting Information

Additional supporting information can be found online in the Supporting Information section. **Supporting Fig. S1:** The class-averaged trends of GT descriptors across weak, medium, and strong PPIs. These descriptors are averaged over the 5 input parameters of training dataset. **Supporting Fig. S2:** Confusion matrix of different protein–protein interaction strengths using feature classification (relabeled). The obtained total accuracy of prediction is 99.83%. **Supporting Table S1:** GT parameters description.