Journal of Imaging Informatics in Medicine
https://doi.org/10.1007/s10278-025-01635-y

Graph Neural Networks for Realistic Bleeding Prediction in Surgical Simulators

Yasar C. Kakdas[1] · Suvranu De[2] · Doga Demirel[3]

Received: 7 April 2025 / Revised: 30 June 2025 / Accepted: 28 July 2025

Abstract
This study presents a novel approach using graph neural networks to predict the risk of internal bleeding using vessel maps derived from patient CT and MRI scans, aimed at enhancing the realism of surgical simulators for emergency scenarios such as trauma, where rapid detection of internal bleeding can be lifesaving. First, medical images are segmented and converted into graph representations of the vasculature, where nodes represent vessel branching points with spatial coordinates and edges encode vessel features such as length and radius. Due to no existing dataset directly labeling bleeding risks, we calculate the bleeding probability for each vessel node using a physics-based heuristic, peripheral vascular resistance via the Hagen-Poiseuille equation. A graph attention network is then trained to regress these probabilities, effectively learning to predict hemorrhage risk from the graph-structured imaging data. The model is trained using a tenfold cross-validation on a combined dataset of 1708 vessel graphs extracted from four public image datasets (MSD, KiTS, AbdomenCT, CT-ORG) with optimization via the Adam optimizer, mean squared error loss, early stopping, and L2 regularization. Our model achieves a mean R-squared of 0.86, reaching up to 0.9188 in optimal configurations and low mean training and validation losses of 0.0069 and 0.0074, respectively, in predicting bleeding risk, with higher performance on well-connected vascular graphs. Finally, we integrate the trained model into an immersive virtual reality environment to simulate intra-abdominal bleeding scenarios for immersive surgical training. The model demonstrates robust predictive performance despite the inherent sparsity of real-life datasets.

Keywords Bleeding prediction · Graph neural network · Surgical simulator · Virtual reality · Medical imaging · Trauma training

Introduction

Internal bleeding, hemorrhage, in trauma patients is a life-threatening condition that must be identified and managed rapidly [1, 2]. Statistics show that trauma is the leading cause of death with over five million people worldwide fatalities due to injuries each year [3]. Massive hemorrhage is the cause of death in 40% of trauma deaths and is considered one of the most life-threatening complications of injuries [2, 4]. Despite its fatal consequences, hemorrhage is treatable once they are detected at the early stages of the injury. Therefore, most of the trauma-based deaths are preventable with appropriate intervention [4–7]. This has motivated the development of advanced training tools, including high-fidelity virtual reality (VR) surgical simulators to help clinicians practice diagnosing and treating internal bleeding in a risk-free environment [8–10]. However, current simulators typically rely on simplistic or pre-programmed models of bleeding, which may not capture the patient-specific variability seen in real-life hemorrhage cases. There is a need for simulation approaches that use actual patient data to drive realistic bleeding behavior, thereby improving training realism and effectiveness.

Meanwhile, the field of medical imaging informatics has produced repositories of clinical imaging data, such as CT and MRI scans, and increasingly powerful AI methods to analyze them. In particular, deep learning has shown great

✉ Doga Demirel
doga@ou.edu

[1] Department of Computer Science, Florida Polytechnic University, Lakeland, FL, USA

[2] College of Engineering, Florida A&M University–Florida State University, Tallahassee, FL, USA

[3] School of Computer Science, University of Oklahoma, Norman, OK, USA

Published online: 12 August 2025

promise in interpreting medical images for diagnosis and risk stratification [11]. Graph neural networks (GNNs) are an emerging class of deep learning models that operate on graph-structured data [12, 13] and are well-suited for representing anatomical networks such as blood vessel structures. Unlike convolutional neural networks (CNNs), which excel at image pixel grids, GNNs can capture complex relational information by modeling anatomical entities as nodes and their connections as edges. GNN approaches have been applied successfully in a variety of biomedical contexts, for example, to analyze brain connectivity graphs in fMRI data [14], to predict patient diagnoses or outcomes from relational health data [15], and to model molecular or drug interaction networks [16–18]. These works suggest that GNNs can extract meaningful patterns from graph representations of medical data. However, to our knowledge, GNNs have not yet been utilized to predict acute pathological events like hemorrhage using medical imaging data. Recent research has begun to bridge physiological modeling with machine learning, using a physics-informed GNN to predict blood flow and pressure in cerebral vessels [19], but such methods have not been extended to simulating emergency scenarios or integrated into training tools.

In this study, we propose a unique framework that combines imaging informatics and GNNs to predict internal bleeding risks and using those predictions in a VR simulator. Our approach begins with extracting detailed maps of blood vessels from computed tomography (CT) and magnetic resonance imaging (MRI) scans of the abdomen. Then, we convert the vessel maps into graphs and compute the bleeding probability of each node based on peripheral vascular resistance (PVR) [20]. By mapping the computed resistance values to a probability scale, we obtain a plausible risk score for hemorrhage at each location. These scores serve as ground truth for training our GNN model. We employ a graph attention network (GAT), a type of GNN that can weigh the importance of neighboring nodes' features to predict the bleeding probability at each node given the graph of the entire vessel tree. The GAT is trained and validated on a combination of real medical imaging datasets with 1708 total graphs.

We integrate the trained GNN into a VR surgical simulator to create an interactive training scenario for intra-abdominal bleeding (hemoperitoneum). In our simulator, the GNN's output, predicted high-risk vessels, is used to determine where and when bleeding should occur in the virtual patient model. This means each simulation run can be personalized to a patient's anatomy and injury risk, which is a substantial improvement over a fixed, scripted scenario. To our knowledge, this is the first time that imaging-derived AI predictions have been used to drive a real-time surgical simulation. Prior works on VR surgical training have implemented bleeding effects with graphical techniques [21], but without using patient-specific data or predictive analytics. By bridging medical image analysis with simulation, our framework allows trainees to encounter more realistic and varied hemorrhage scenarios, informed by actual clinical data.

Materials and Methods

Our framework, as depicted in Fig. 1, consists of four major components: a) pre-processing medical image datasets consisting of binary segmented CT and MRI scans into graphs that encapsulate the vascular tree extracted from images

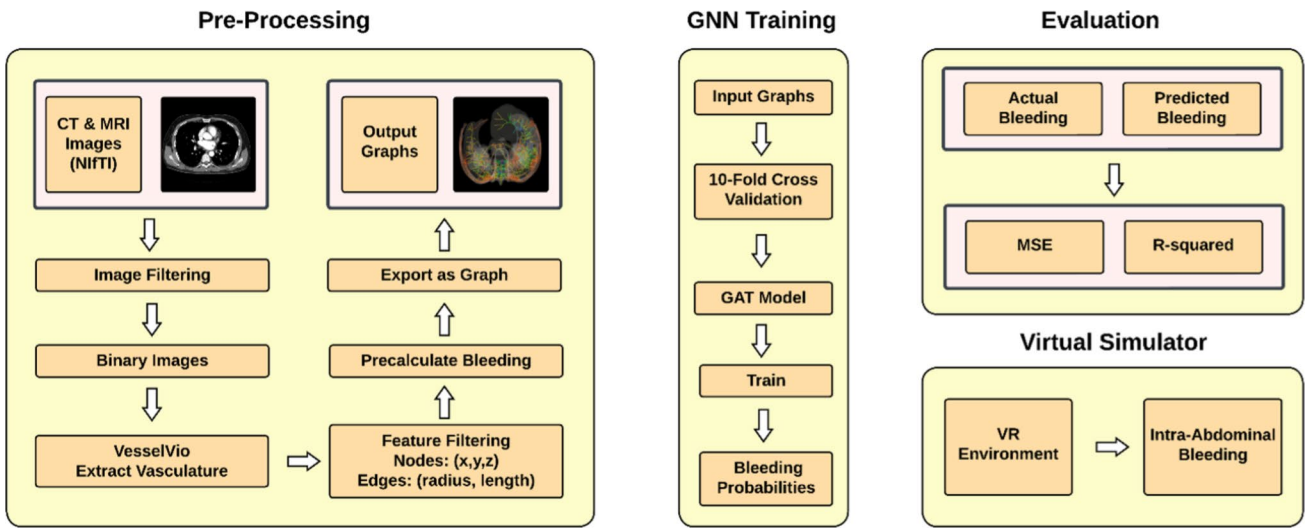

Fig. 1 System diagram of the framework

where nodes are branching points with (x, y, z) coordinates metadata and edges are vessels with several features such as radius and length of the vessel; b) a GNN-based node regression model that predicts the bleeding probability at each vessel node; c) evaluation of the model's performance considering training, validation losses, and R-squared; d) deployment of the framework into a virtual scene by building a bleeding-based medical scenario, in our case, intra-abdominal bleeding [22].

Data Pre-processing

The first part of our framework pipeline is to find appropriate CT and MRI scan images and process them. We have used public datasets from the Medical Segmentation Decathlon (MSD) [23], Kidney Tumor Segmentation Challenge (KiTS23) [24], AbdomenCT-1 K [25], and CT-ORG [26]. From these, we extracted the blood vessels in organs and the abdominal cavity by selecting the segmentation labels corresponding to vascular structures. All images retrieved from the datasets are encoded in the Neuroimaging Informatics Technology Initiative (NIfTI) [27] file format.

We used a framework named VesselVio [28] for the vasculature analysis. VesselVio is an open-source application that analyzes and visualizes vasculature datasets. The VesselVio only supports binarized images. Therefore, the first step of our preprocessing involves filtering non-binary images. After filtering the data, we used VesselVio and generated graph files containing the vasculature's metadata. In a graph, we have nodes representing the branching and end points of the vessels with the coordinates of (x, y, z). The edges are the vessels with the features of average radius and the vessel's length. The metadata of the dataset after preprocessing can be seen in Table 1.

The next step is to calculate bleeding probabilities. To the best of our knowledge, there is no ready-to-use dataset that classifies the bleeding probabilities of a vessel structure. Therefore, to be able to train our model, we generated bleeding probability labels for training in a heuristic but physiologically informed way by considering the diameter and length of the vessel. For this calculation, we used PVR [20]. PVR refers to the resistance that blood encounters while flowing through vessels. Elevated PVR often leads to increased blood pressure and eventually contributes to hypertension [29, 30]. High blood pressure (hypertension) can weaken vessel walls and develop varices that are prone to rupture and cause bleeding [31–33]. PVR is calculated by the Hagen-Poiseuille Eq. (1) [29, 34, 35].

$$R = 8l\eta/\pi r^4 \tag{1}$$

where R represents resistance, l represents vessel length, η represents blood viscosity, and r represents vessel radius. As seen in Eq. (1), the resistance is directly proportional to the ratio of length and fourth power of the radius. For each node, we traverse its connected edges, i.e., vessels, and use the mean length and radius of all neighbors in Eq. (1). This yielded a resistance score for each node. We then normalized these scores across each graph to a 0–1 range to obtain a bleeding probability, p, for each node. Intuitively, nodes connected to longer and thinner vessels receive p closer to 1 (higher risk), whereas nodes in short or wide vessels get p near 0 (lower risk). Finally, these probability values were attached as a node feature/label in the graph. After this step, each graph's nodes have features of (x, y, z, p), and each edge has features of *length* and *radius*. This creates the training data for our GNN. This method of labeling is a proxy; however, it provides a consistent, physiology-based way to label a large dataset without manual annotation, while embedding domain knowledge from Eq. (1) that the GNN can learn to replicate or refine.

Table 1 Dataset metadata

Dataset source	# of images	# of nodes	# of edges
MSD	933	61,460	56,742
KiTS	487	46,166	49,172
AbdomenCT	288	87,748	100,129
CT-ORG	37	159,384	207,104

Graph Attention Network (GAT) Bleeding Prediction

We have a total of 1745 graphs, 354,758 nodes, and 413,147 edges to process. We used GNN for the node regression task. The model tries to predict the probability of bleeding as a node feature. Due to using different datasets from multiple resources, the difference between the graphs in terms of size is significantly large. To tackle this issue, the k-fold cross-validation ($k = 10$) [36] technique was used. k-fold cross-validation shuffles the dataset randomly, splits it into k-groups, and picks one group as the validation set and the rest as the training set. This procedure is applied for each fold. Hence, the model is trained by a k-different dataset partition, overcoming the potential for overfitting and providing more robust results.

For the prediction logic, we used GAT, a neural network architecture designed to process graph-structured data. Unlike traditional GCNs with fixed aggregation methods, GATs use attention mechanisms to weigh the importance of neighboring nodes adaptively and allow nodes to weigh more on relevant neighbors and less on irrelevant ones [37]. This model suits our dataset since the vessel maps are not necessarily a single big graph with each node connected. Instead, we might end up with many independent small graphs due to the nature of extracting from real

scanning images. Therefore, passing messages with relevant neighbors while updating the nodes increases the accuracy of our implementation.

As demonstrated in Fig. 2, our custom GAT model architecture consists of two convolution layers designed to leverage the graph attention mechanism. The first layer takes the input dimension of node features, the dimension of the hidden layer, and the number of attention heads, and outputs by concatenating the results from different attention heads. In this implementation, we employed four attention heads with a hidden dimension of 64, a choice derived from hyperparameter tuning. The use of four attention heads enhances the model's capability to capture a more comprehensive array of relationships between nodes. Additionally, concatenation ensures that diverse patterns identified by separate heads are effectively integrated and passed to subsequent layers. The second convolution layer processes the concatenated outputs, but instead of further concatenation, it averages the contributions from each attention head, effectively reducing the output dimension to the desired output size. The forward function processes input through the first GAT convolution layer and applies the rectified linear unit (ReLU) [38] activation function to its output. This choice of activation function introduces non-linearity, enabling the model to learn more complex patterns. The activated outputs are then processed by the second GAT convolution layer, which averages across heads without applying an additional activation function, allowing to stabilize the training process.

The training loop utilizes the Adam optimizer [39] and mean squared error (MSE) loss function. The training dataset is traversed for each epoch, a prediction is calculated with MSE loss, and the loss is backpropagated to update the model's weights. After the update, the validation dataset is used to evaluate the model's performance. Rather than employing a straightforward train-test split, our methodology utilizes *k*-fold cross-validation. This approach ensures the model is evaluated across diverse data subsets, offering a more robust indication of its generalizability and effectiveness.

Early stopping is implemented to prevent overfitting. The training halts if there is no improvement in validation loss over a designated number of epochs, ensuring the model is captured at its optimal state. Furthermore, weight decay is incorporated with the Adam optimizer to introduce L2 regularization, discouraging overfitting by penalizing large weights.

A critical aspect of our model is its detailed attention to graph data, specifically through the handling of node features, edge attributes, and edge indices. These elements are crucial for the convolution layers, which leverage node features and edge attributes alongside the structural information provided by edge indices to adjust the attention mechanisms dynamically. This approach allows the model to efficiently learn from the intricate relationships present within graph data, providing valuable insights into the underlying patterns and interactions.

Hyperparameter Tuning and Evaluation Metrics for the Designed Model

The goal is to predict continuous values representing the chances of bleeding from 0 to 1. Therefore, our task is a regression problem. MSE was used as the loss function during training and evaluation. MSE is commonly used for regression problems. It calculates the average of the squared differences between the predicted values and the actual ground truth values. During the training, the optimizer tries to minimize MSE by tuning model parameters. Meanwhile, during the evaluation, MSE quantitatively measures how well the model predicts. Different hyperparameters, such as the number of hidden layers, attention heads, and epochs, were determined based on the comparison of the MSE loss between the runs during the tuning. Based on preliminary trials, we selected four attention heads and a 64-dimension hidden layer for the first GAT layer. We observed that fewer heads reduced the model's ability to capture complex patterns, whereas more than four heads resulted in lower accuracy while increasing complexity. Similarly, a hidden

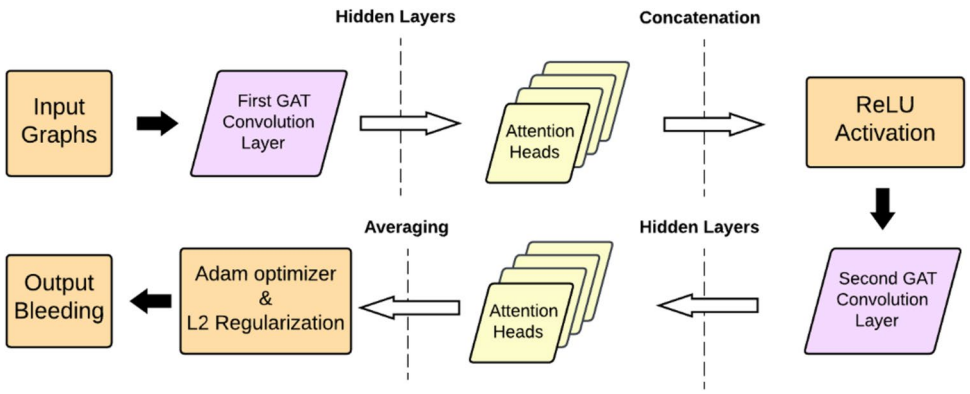

Fig. 2 Graph attention network architecture

dimension of 64 offered a good balance, where fewer than 32 dimensions caused underfitting and 128 dimensions gave marginal gains. We applied L2 weight regularization with a weight decay of $1e^{-4}$ to all GNN weights during training to prevent overfitting. We used a 0.001 learning rate to ensure stable training. To validate the hyperparameter selections, we carried out an ablation study in the "Ablation Study: Hyperparameter Sensitivity" section. Additionally, *R*-squared [40], also known as the coefficient of determination, was reported after each training. *R*-squared indicates how well the model replicates the observed outcomes and how well the model has the ability to capture variability.

Case Study: Intra-Abdominal Bleeding Virtual Reality Simulator

We implemented a flexible framework that can quickly adapt to other datasets as long as the images are segmented and binarized. The model can also be easily tuned up with different nodes and edge features. We implemented a VR-based intra-abdominal bleeding scenario as a case study. The scenario is built using Unity3D game engine (Unity 2022.3.21f1) and runs on a Meta Quest 2 head-mounted display for immersion. In the scene, there is an operating room with an intubated patient with ready-to-use laparoscopic instruments. This scene supports multiplayer functionality, allowing multiple participants to simultaneously employ various laparoscopic instruments such as a grabber, camera, cauterizer, trocar, and suction. To enhance the realism of the simulation, we have integrated high-fidelity organ models. This abdominal anatomy model consists of the colon, spleen, liver, stomach, pancreas, small intestines, gall bladder, bile duct, and esophagus. Each organ has been designed with a high mesh count along with detailed, high-resolution texture and normal maps to increase the fidelity of the simulation. Another feature we have included in this simulator to increase the realism is the custom-implemented bleeding effects. To achieve the bleeding effects, we implemented a custom shader by using the high-definition render pipeline (HDRP) unlit shader graphs. In this shader graph, the distortion effect was used to mimic the waves of the liquid. To create the distortion effect, the UV map of a 2D noise texture was tiled and offset over time to provide a scrolled texture. Then, the scrolled texture was added to the screen position, and the exposure with scene color was used to create the distortion effect. Moreover, the density of the distortion was implemented as a variable to create different bleeding effects for different scenarios.

The key link between the GNN model and the VR simulator is through the vessel graph predictions. At the start of a simulation session, our framework takes either a pre-loaded patient scan or a user-selected CT scan, processes it into a vessel graph, and then feeds it into the trained GNN model to obtain predicted bleeding probabilities for each vessel node. These probabilities are then used to probabilistically determine which region will bleed during the simulation. In practice, we select one or a few of the highest-risk nodes as the bleeding source. By default, if no user scan is provided, the simulator uses a representative vessel graph from our dataset and the corresponding precomputed risk predictions.

The VR engine then maps the chosen vessel node(s) to the 3D organ model. We achieve this by aligning the coordinate system of the graph (which was originally in image voxel space) with the coordinate system of the 3D patient model in Unity. Since both are spatial representations of the anatomy, this mapping is possible via a linear transform (we calibrate using landmarks visible in the CT and the model, such as organ boundaries). Once mapped, a node's (x, y, z) position corresponds to a location inside a specific organ in the virtual patient. Sample mapping of vessel structure to an organ is demonstrated in Fig. 3.

When the simulation scenario begins in the operation room, they are given a brief on the patient's history regarding the results obtained from the framework, so that the trainees will have insight into where the bleeding could happen. The first indication of the scenario is seen when the patient's vital signs deteriorate, and vital monitors start to alert. In each frame, the displays for vital signs, including a large screen on the wall and a smaller one on the anesthesia

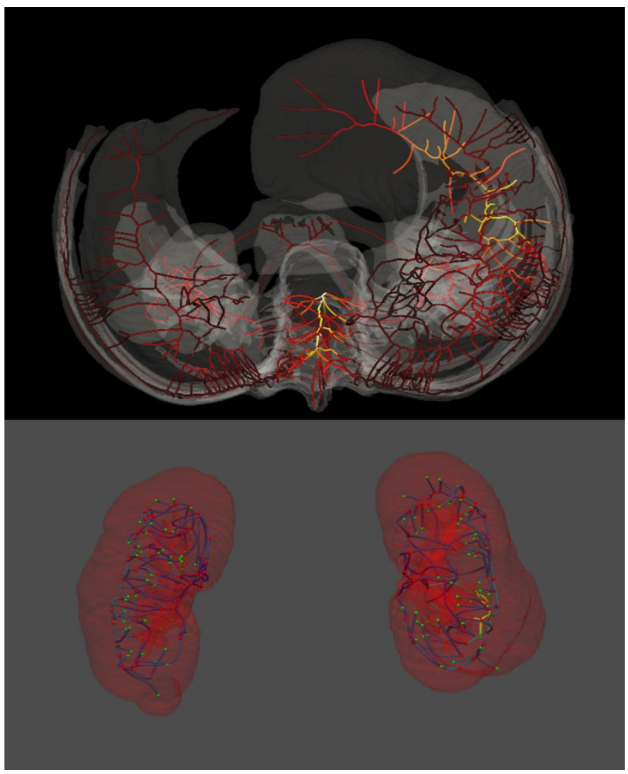

Fig. 3 Vessel structure to 3D model mapping

machine's arm, are continuously updated by the pulse physiology engine [41].

The GNN's role is essentially behind the scenes, as it determines where the bleed occurs. Each time the simulation is reset or a new patient's scan is loaded, a different location or vessel can be chosen based on the risk profile, making every scenario unique and challenging. By integrating the GNN in this manner, the simulator provides a realistic experience: the bleeding is not random but based on an underlying risk model derived from real imaging data. The simulator also logs whether the trainees manage to stop the bleeding for training assessment purposes. When the trainee(s) suspect internal bleeding based on changes in vital signs in VR, they utilize laparoscopic instruments to identify and control the bleeding. One of the trainees(s) deploys a laparoscopic camera to visually inspect the abdominal cavity and identify the source of the bleeding. Throughout the procedure, two screens display real-time images captured by the camera. Once the bleeding source is located, one of the trainees(s) employs a suction device to extract the blood. Both trainees' perspectives during the simulation are presented in Fig. 4. The overview of the whole scene can be seen in Fig. 5.

Fig. 5 Overview of the virtual abdominal bleeding scene

Results

In our comprehensive study, we employed GNN with GAT, aiming to predict bleeding from real-life medical imaging data. Acknowledging the sparse nature of real-life datasets, we utilized a tenfold cross-validation strategy to enhance the model's robustness and generalizability. We standardized the number of training epochs to 100 for each fold, implementing early stopping with a patience parameter set to 10 epochs to prevent overfitting. Optimization was achieved through the Adam optimizer and L2 regularization, setting a weight decay of 0.0001. The GAT architecture featured four attention heads and 64 hidden dimensions, with a learning rate of 0.001.

The training process was evaluated over many epochs, and the performance was assessed through the MSE loss function for both training and validation datasets, determined by the *k*-fold cross-validation. Additionally, *R*-squared was used as a metric to determine the proportion of variance in the dependent variable that could be predicted from the independent variable. Training occurred separately for each dataset detailed in the "Data Pre-processing" section, including a combined dataset approach, with parameters consistently maintained across all instances for fair comparison. Following each fold, parameters were reset to their original states, with losses and *R*-squared values recorded at each epoch.

All model training and experiments were performed on an Intel i7-11800H processor with NVIDIA GeForce RTX 3070 GPU (8 GB GDDR6) and 16 GB of RAM. Across all 10 folds and all datasets, the total training time was 8 h. Throughout training, the GPU memory usage stayed below 4 GB. Also, we measured the model's inference speed. After training, we measured the trained GNN model's inference speed for vessel graphs of varying sizes. The trained GNN processed typical vessel graphs and produced bleeding probabilities in 9–28 ms for graphs with up to 500 nodes, 28–50 ms for graphs with 500–1000 nodes, and 50–130 ms for graphs with 1000–5000 nodes.

The next subsections report each dataset's performance along with the combined dataset's overall performance, and a hyperparameter sensitivity ablation study. Figures 6, 7, 8, and 9 visually depict training and validation losses per epoch for the most successful folds across datasets and *R*-squared results for each dataset, illustrating the performance and learning dynamics of our GNN model.

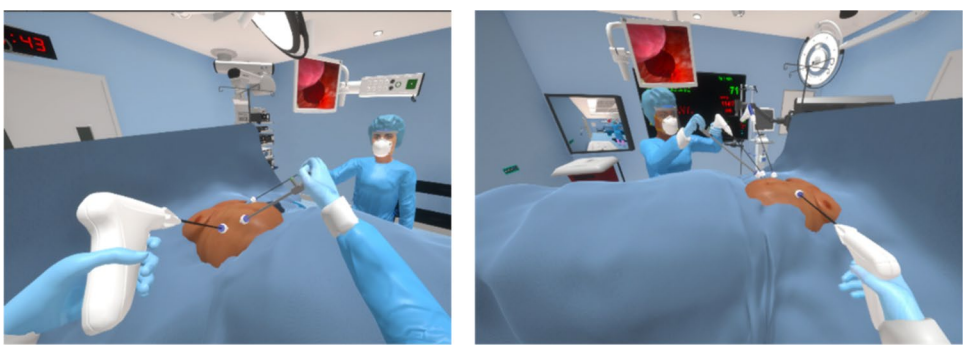

Fig. 4 Clinicians' perspective during the simulation

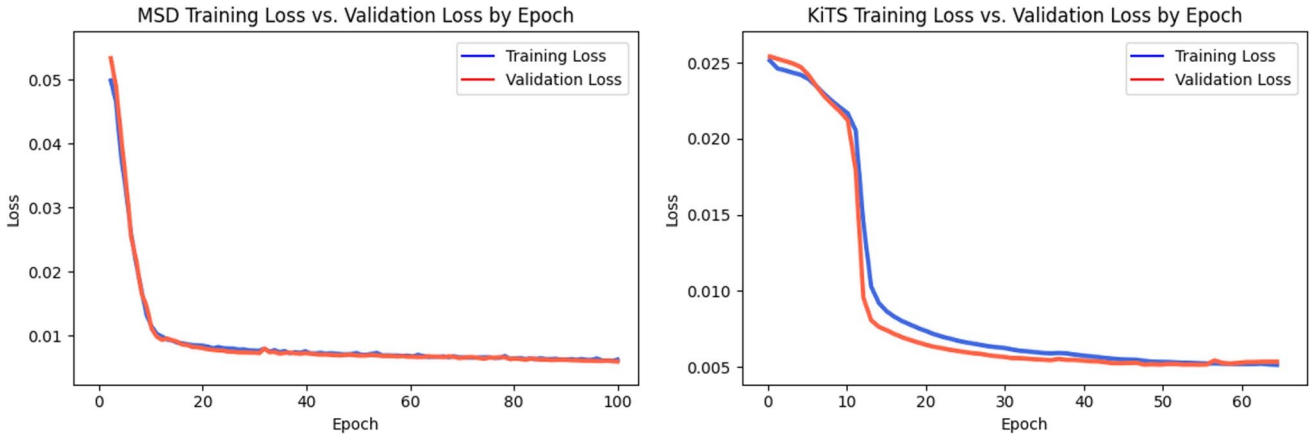

Fig. 6 Graph attention network performance for MSD and KiTS datasets

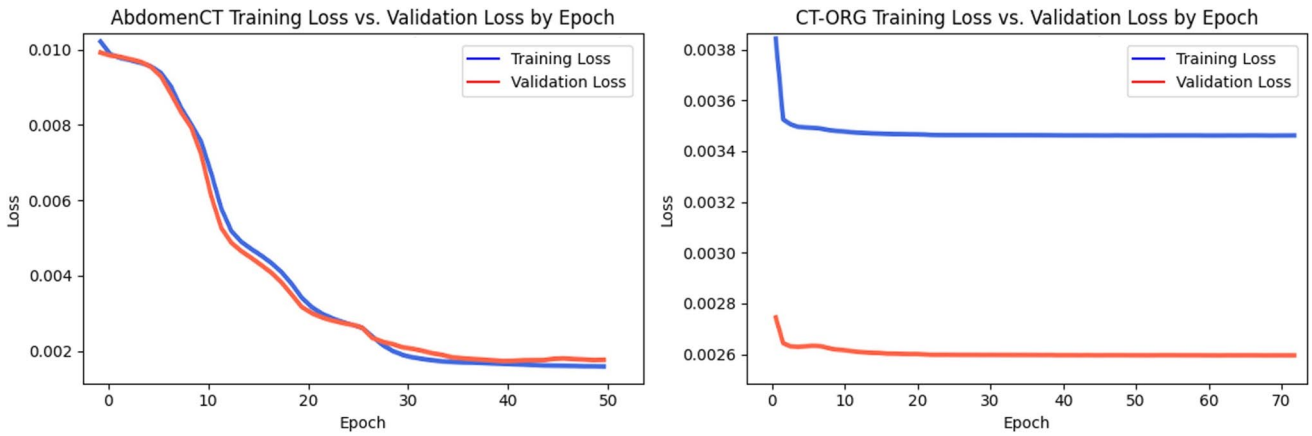

Fig. 7 Graph attention network performance for AbdomenCT and CT-ORG datasets

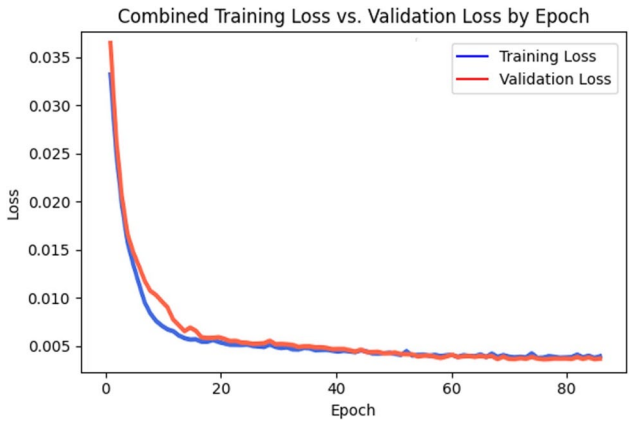

Fig. 8 Graph attention network performance for the combined dataset

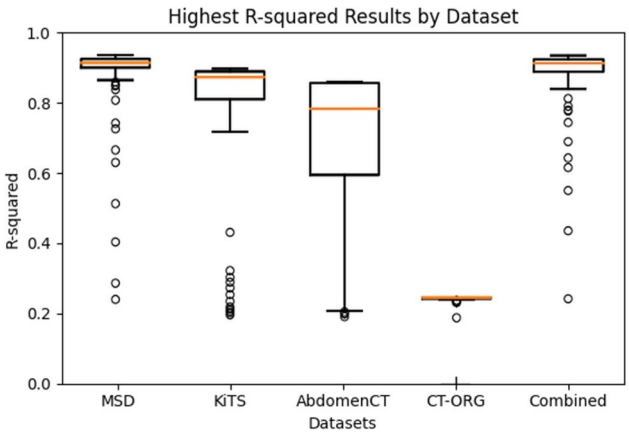

Fig. 9 Highest *R*-squared values achieved across different datasets

Performance of the MSD Dataset

The MSD dataset contained 933 graphs with a total of 61,460 nodes and 56,742 edges. These graphs had an average density of 0.0198, ranging from 0.0089 to 1.0 with a standard deviation of 0.2751. Over tenfold cross-validation, the model achieved a mean training loss of 0.0098 and a validation loss of 0.0101 across all folds. The best fold resulted in a notably lower training and validation losses of 0.0050 and 0.0047 within 100 epochs, respectively, as seen in Fig. 6. The average R-squared across folds was 0.8963, with the best R-squared of 0.9216, indicating that the model explained approximately 89.6 to 92.2% of the variance in bleeding probability labels. Training on the MSD dataset concluded in an average of 89.6 epochs, ranging from 39 to 100. The standard deviation of losses across folds was low, with 0.0022 and 0.0024 for training and validation losses, respectively, indicating stable learning.

Performance of the KiTS Dataset

The KiTS dataset had 487 graphs with 46,166 nodes and 49,172 edges. Compared to the MSD dataset, KiTS had a higher average graph density of 0.0379, ranging from 0.0048 to 0.2857 with a standard deviation of 0.0352. The model showed mean training and validation losses of 0.0079 and 0.0077, respectively. As shown in Fig. 6, the best fold achieved a lower training loss of 0.0046 and validation loss of 0.0048 over 66 epochs. R-squared scores were similarly high, with a peak of 0.9147 and a mean of 0.8784 across folds. Training was completed in an average of 86.4 epochs, spanning from 43 to 100. The variability in losses was low, and standard deviations for training and validation losses were 0.0019 and 0.0016, respectively, showing stable, consistent performance.

Performance of AbdomenCT Dataset

The AbdomenCT dataset had larger but sparser graphs. This dataset consists of 288 graphs with 87,748 nodes and 100,129 edges, with an average density of 0.0103 (ranging from 0.0012 to 0.0250 with a standard deviation of 0.0047). According to Fig. 7, the GNN achieved mean training and validation losses of 0.0039 and 0.0038, respectively, indicating smaller error magnitudes. The best fold reduced the training loss to 0.0017 and the validation loss to 0.0018 in 51 epochs. Standard deviations for training and validation losses were 0.0014 and 0.0013, respectively. The average R-squared was 0.8283, peaking at 0.8670, which was lower than MSD and KiTS. On average, training took 89.1 epochs, with a range from 48 to 100.

Performance of CT-ORG Dataset

With 37 graphs, 159,384 nodes, and 207,104 edges, the extremely sparse CT-ORG dataset had an average density of 0.0010 (ranging from 0.0001 to 0.0024 with a standard deviation of 0.0005). This dataset was an outlier in that each graph is a huge network covering many organs, but connections are sparse. The model's performance on CT-ORG was weaker relative to other sets with mean training and validation losses of 0.0034 and 0.0035, respectively, as illustrated in Fig. 7. The standout fold showed a training loss of 0.0034 and a validation loss of 0.0026 in 71 epochs. The standard deviations for training and validation losses were 0.0001 and 0.0011, respectively. The highest R-squared obtained on CT-ORG was only 0.3859, and the average across folds was 0.2700, showing that the model struggled to capture the variability in this dataset. Training completed early, in an average of 30.5 epochs, ranging from 13 to 71, as the model would hit a plateau or start overfitting quickly.

Performance of Combined Dataset

We also evaluated the model on the combined dataset. Combining all datasets resulted in 1708 graphs with 354,758 nodes and 413,147 edges. Training on the combined data tests the model's ability to generalize across all types of anatomy and imaging sources. The average density of the combined dataset was 0.1182 (ranging from 0.0001 to 1.0 with a standard deviation of 0.2196). For the combined dataset, Fig. 8 shows that the mean training and validation losses across all folds were strong with 0.0069 and 0.0074, respectively. The best fold had a training loss of 0.0042 and a validation loss of 0.0040 in 87 epochs. The standard deviations for training and validation losses across folds were 0.0011 and 0.0018, respectively. The best R-squared observed on the combined validation was 0.9188 and a mean of 0.8600. Training completed in an average of 76.9 epochs, spanning 50 to 100.

Ablation Study: Hyperparameter Sensitivity

To support our design choices and assess model robustness, we performed a sensitivity analysis on two key hyperparameters: the number of attention heads in the GAT layers and the use of L2 regularization weight decay. We used the combined dataset for these experiments to ensure sufficient data for evaluation. We trained variant models under different settings and compared their performance using the same tenfold cross-validation, with results averaged.

For attention heads, we tested models with two, four, and eight heads in each GAT layer, adjusting the hidden dimension per head such that the total output dimension

of the first layer remained roughly constant at 256, to keep model capacity comparable. As reported previously, a four-head model was used for this study with an average *R*-squared of 0.86 and stable training. With two heads, the model converged but showed a small drop in accuracy to 0.82. Validation MSE increased by about 5–10% relative to the four-head model. Also, we noticed slightly higher variance in performance across folds, suggesting two heads might under-represent some relationships. For eight heads, the model's performance was slightly lower with an average *R*-squared of 0.85. However, training with eight heads was 20–25% slower per epoch, and in 10% of the time, it required more epochs to converge fully.

The second hyperparameter we performed a sensitivity analysis on was L2 regularization. We examined the effect of L2 regularization weight decay by training a model with weight decay set to no regularization, compared to 0.0001. Without L2 regularization, the final training loss was 13.2% lower than the regularized model, but the validation loss was 20% higher MSE than the regularized model. The validation *R*-squared dropped to 0.80 on average. In 10% of the folds, the non-regularized model's validation *R*-squared was between 0.70 and 0.78. The training curves for the unregularized model often had the validation loss starting to increase after a certain number of epochs, whereas the regularized model's validation loss stayed more aligned with training loss. This confirms the model was overfitting when regularization was removed.

Discussion

The "Results" section showcases the performance of the GNNs across different datasets, providing insights through loss metrics and *R*-squared values that reflect the model's accuracy and predictive power. In general, the model demonstrated strong predictive accuracy across datasets with consistently high *R*-squared values, showing its ability to generalize patterns within varying vascular complexities.

The performance analysis across various datasets provides insights into how graph density, illustrated in Fig. 10, and dataset characteristics influence model outcomes regarding loss and *R*-squared values. The MSD dataset, which has an average density of 0.0198, allowed the model to achieve high predictive accuracy with the best *R*-squared value of 0.9216. Also, we observed a mean training loss of 0.0098 and a validation loss of 0.0101. These metrics indicate a strong model performance, effectively balancing error minimization and predictive accuracy. The moderate density of the MSD dataset likely facilitates this by providing sufficient information without overwhelming the model, a critical factor in achieving high generalization capabilities. MSD dataset, despite the variability in graph complexity related to different organ vessels, the model's consistent performance indicates it effectively leveraged sufficient connectivity to generalize without overfitting.

Transitioning to the KiTS dataset, with its higher density of 0.0379, we see a continuation of this trend with even slightly improved loss values (mean training and validation losses of 0.0079 and 0.0077, respectively) and a comparable *R*-squared value peaking at 0.9147. This suggests that increased graph density facilitated better relational feature learning, enhancing the model's ability to accurately predict risk patterns within specialized anatomical regions.

The AbdomenCT and CT-ORG datasets presented unique challenges due to their varying densities and complexities. Despite its lower density of 0.0103, the AbdomenCT dataset shows the lowest mean training and validation losses compared to the MSD and KiTS datasets but with a slightly lower *R*-squared value of 0.8670. This suggests that while sparse datasets might improve error minimization, they could limit the model's ability to accurately capture and

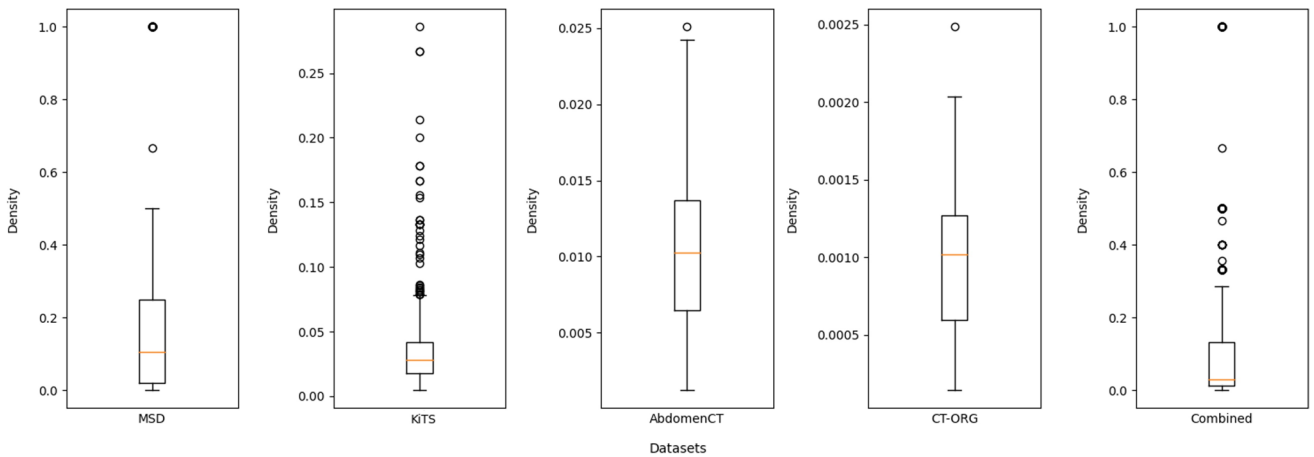

Fig. 10 Graph density comparison across datasets

predict more complex patterns. Similarly, the CT-ORG dataset, with the lowest density of 0.0010 among all datasets, shows good results in terms of training and validation losses (0.0034 and 0.0035, respectively), but the *R*-squared values significantly drop, with the best at only 0.3859. This dataset suffered from overfitting and underfitting across all folds. Its lower *R*-squared indicates the model's inability to capture complex patterns and variability. The GNN requires structural information to effectively propagate and utilize relational data, highlighting the importance of graph density and connectivity in predictive modeling tasks.

In contrast, the combined dataset, with an average density of 0.1182, benefits from the structural advantages and challenges of the individual datasets, resulting in mean training and validation losses of 0.0069 and 0.0074, respectively, and a robust best *R*-squared value of 0.9188. This demonstrates the importance of combining diverse datasets to create a rich, varied learning environment that mitigates the limitations of individual, real-life, sparse datasets.

The ablation study further confirmed the robustness of the chosen model architecture. Optimal performance was achieved with four attention heads and L2 regularization, demonstrating that moderate model complexity and appropriate regularization significantly improve predictive generalizability and stability.

Overall, these results emphasize that while GNNs can achieve strong performance in predicting bleeding risk from anatomical graphs, their effectiveness is influenced significantly by the density and structural properties of the data. Although providing a GNN model that performs well on various datasets is challenging, our proposed model showed its capability to capture complex patterns with different characteristics for different datasets. While high-density datasets generally support better learning outcomes, our approach shows promising results for extremely low densities. This approach mitigates the challenges encountered with real-life datasets. In our current workflow, bleeding risk prediction is performed once prior to the start of the simulation for each patient-specific anatomy. Therefore, while our GNN model is capable of near real-time inference on modern hardware, real-time execution is not strictly required for end-user experience in this application. The critical factors are accuracy and reproducibility of the risk assessment, which enable realistic and safe simulation scenarios. However, our framework encountered some limitations. During the preprocessing stage, VesselVio, which was used to extract vessel structure to graphs, only accepts pre-segmented binary CT and MRI scan images. Therefore, we had to eliminate non-binary images. This limited the number of samples we could use during the training. Another limitation of our current approach is the reliance on the Hagen–Poiseuille assumption, which considers ideal cylindrical vessels and steady flow. Real human vasculature is more complex (non-uniform cross-sections, branching flows, etc.). While our risk labeling heuristic captures the riskier intuition, it may not account for other factors like vessel wall integrity or surrounding tissue support. This means our GNN is learning a simplified representation of bleeding risk. This is acceptable for simulation training as we are not predicting actual patient outcomes, but it means the model might not directly translate to a clinical diagnostic tool without further validation or augmentation. In the future, incorporating patient-specific variables such as blood pressure and coagulopathy status into the model could make the risk predictions more personalized.

Conclusion

This study demonstrates the application of GNNs for predicting internal bleeding risk from medical imaging data and highlights the significant potential of this approach in enhancing the realism and effectiveness of surgical simulations. Our framework utilizes GATs for node regression tasks within the vascular structures derived from real-life CT and MRI scan images. The evaluation of our model's performance, conducted through comprehensive testing and validated by a tenfold cross-validation, underscores the accuracy and reliability of our framework.

The combination of various datasets with different characteristics, 1708 graphs with over 354,748 nodes and 413,147 edges, demonstrated promising results with mean training loss and validation loss of 0.0069 and 0.0074, respectively. Furthermore, the high *R*-squared score of 0.9188 reflects our model's accuracy in predicting bleeding and its capacity to generalize across diverse medical imaging datasets. Our model's ability to adapt to different datasets is essential for developing surgical simulations that can be applied to various clinical scenarios. Moreover, the minimal deviation between training and validation losses highlights the model's stability and predictive consistency, which are crucial for its implementation in real-world applications.

Additionally, as a case study, we developed a VR simulator for an intra-abdominal bleeding scenario to demonstrate the practical integration of our GNN predictions. The integration of GAT bleeding probabilities within an immersive VR environment provided a unique opportunity to train and test surgical skills accurately in a risk-free environment. This connection of AI-based risk modeling with interactive simulation allowed each run of the simulator to present a slightly different scenario driven by patient-specific data, thereby increasing the training value.

The potential for further advancement of the model to include more complex bleeding scenarios, coupled with the integration of real-time patient data or sensor feedback into the model, presents a promising avenue for

creating engaging, effective, and realistic medical simulations. Another promising avenue is to validate and refine the model using actual clinical cases of hemorrhage, which could improve its fidelity beyond the current heuristic-based labeling. Even in its present form, our framework presents a promising step toward creating engaging, effective, and data-driven realistic medical simulations. It illustrates how vascular risk predictions from medical imaging analysis can directly inform simulation training, ultimately bridging the gap between computational modeling and hands-on medical education.

Author Contribution Yasar C. Kakdas: conceptualization, methodology, software, validation, formal analysis, writing—original draft, writing—review and editing, and visualization. Suvranu De: conceptualization, investigation, resources, supervision, and funding acquisition. Doga Demirel: conceptualization, methodology, validation, investigation, resources, writing—original draft, writing—review and editing, supervision, project administration, and funding acquisition.

Funding This project was supported by grants from the National Institutes of Health (NIH)/NIBIB R01EB025241, R01EB033674, R01EB032820, and R01EB005807.

Declarations

Ethics Approval This study did not require ethics approval.

Competing Interests The authors declare no competing interests.

References

1. N. Ahmed, D. Kassavin, Y.-H. Kuo, and R. Biswal, "Sensitivity and specificity of CT scan and angiogram for ongoing internal bleeding following torso trauma," *Emerg. Med. J.*, vol. 30, no. 3, pp. e14–e14, Mar. 2013, https://doi.org/10.1136/emermed-2011-200376.
2. D. R. Spahn *et al.*, "The European guideline on management of major bleeding and coagulopathy following trauma: fifth edition," *Crit. Care*, vol. 23, no. 1, Dec. 2019, https://doi.org/10.1186/s13054-019-2347-3.
3. J. M. M. van Breugel, M. J. S. Niemeyer, R. M. Houwert, R. H. H. Groenwold, L. P. H. Leenen, and K. J. P. van Wessem, "Global changes in mortality rates in polytrauma patients admitted to the ICU—a systematic review," *World J. Emerg. Surg.*, vol. 15, no. 1, p. 55, Sep. 2020, https://doi.org/10.1186/s13017-020-00330-3.
4. P. Rhee *et al.*, "Increasing Trauma Deaths in the United States," *Ann. Surg.*, vol. 260, no. 1, p. 13, Jul. 2014, https://doi.org/10.1097/SLA.0000000000000600.
5. D. R. Spahn *et al.*, "The European guideline on management of major bleeding and coagulopathy following trauma: fifth edition," *Crit. Care*, vol. 23, no. 1, p. 98, Dec. 2019, https://doi.org/10.1186/s13054-019-2347-3.
6. P. M. Cantle and B. A. Cotton, "Prediction of Massive Transfusion in Trauma," *Crit. Care Clin.*, vol. 33, no. 1, pp. 71–84, Jan. 2017, https://doi.org/10.1016/j.ccc.2016.08.002.
7. C. Guo *et al.*, "A prediction model for massive hemorrhage in trauma: a retrospective observational study," *BMC Emerg. Med.*, vol. 22, no. 1, p. 180, Nov. 2022, https://doi.org/10.1186/s12873-022-00737-y.
8. M. Paschold, T. Huber, S. R. Zeissig, H. Lang, and W. Kneist, "Management of bleeding complications in virtual reality laparoscopy," *Int. Surg.*, vol. 104, no. 5–6, pp. 277–282, 2019.
9. R. Sweet, J. Porter, P. Oppenheimer, D. Hendrickson, A. Gupta, and S. Weghorst, "*Third Prize*: Simulation of Bleeding in Endoscopic Procedures Using Virtual Reality," *J. Endourol.*, vol. 16, no. 7, pp. 451–455, Sep. 2002, https://doi.org/10.1089/089277902760367395.
10. U. Erden, M. A. Gromski, S. De, and D. Demirel, "Preliminary validation of the virtual bariatric endoscopic simulator," *iGIE*, vol. 3, no. 4, pp. 453–462, 2024.
11. T. Davenport and R. Kalakota, "The potential for artificial intelligence in healthcare," *Future Healthc. J.*, vol. 6, no. 2, pp. 94–98, Jun. 2019, https://doi.org/10.7861/futurehosp.6-2-94.
12. J. Zhou *et al.*, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, Jan. 2020, https://doi.org/10.1016/j.aiopen.2021.01.001.
13. F. Scarselli, M. Gori, Ah Chung Tsoi, M. Hagenbuchner, and G. Monfardini, "The Graph Neural Network Model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009, https://doi.org/10.1109/TNN.2008.2005605.
14. X. Li *et al.*, "BrainGNN: Interpretable Brain Graph Neural Network for fMRI Analysis," *Med. Image Anal.*, vol. 74, p. 102233, Dec. 2021, https://doi.org/10.1016/j.media.2021.102233.
15. Y. Li, B. Qian, X. Zhang, and H. Liu, "Graph Neural Network-Based Diagnosis Prediction," *Big Data*, vol. 8, no. 5, pp. 379–390, Oct. 2020, https://doi.org/10.1089/big.2020.0070.
16. T. Ma, C. Xiao, J. Zhou, and F. Wang, "Drug Similarity Integration Through Attentive Multi-view Graph Auto-Encoders," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, Stockholm, Sweden: International Joint Conferences on Artificial Intelligence Organization, Jul. 2018, pp. 3477–3483. https://doi.org/10.24963/ijcai.2018/483.
17. M. Zitnik, M. Agrawal, and J. Leskovec, "Modeling polypharmacy side effects with graph convolutional networks," *Bioinformatics*, vol. 34, no. 13, pp. i457–i466, Jul. 2018, https://doi.org/10.1093/bioinformatics/bty294.
18. C. Mao, L. Yao, and Y. Luo, *MedGCN: Graph Convolutional Networks for Multiple Medical Tasks*. 2019.
19. A. Sen, E. Ghajar-Rahimi, M. Aguirre, L. Navarro, C. J. Goergen, and S. Avril, "Physics-Informed Graph Neural Networks to solve 1-D equations of blood flow," *Comput. Methods Programs Biomed.*, vol. 257, p. 108427, 2024.
20. F. J. Haddy, H. W. Overbeck, and R. M. Daugherty, "Peripheral Vascular Resistance," *Annu. Rev. Med.*, vol. 19, no. 1, pp. 167–194, 1968, https://doi.org/10.1146/annurev.me.19.020168.001123.
21. T. Halic, G. Sankaranarayanan, and S. De, "GPU-based efficient realistic techniques for bleeding and smoke generation in surgical simulators," *Int. J. Med. Robot.*, vol. 6, no. 4, pp. 431–443, Dec. 2010, https://doi.org/10.1002/rcs.353.
22. M. Lubner *et al.*, "Blood in the Belly: CT Findings of Hemoperitoneum," *RadioGraphics*, vol. 27, no. 1, pp. 109–125, Jan. 2007, https://doi.org/10.1148/rg.271065042.
23. M. Antonelli *et al.*, "The Medical Segmentation Decathlon," *Nat. Commun.*, vol. 13, no. 1, Art. no. 1, Jul. 2022, https://doi.org/10.1038/s41467-022-30695-9.
24. N. Heller *et al.*, "The state of the art in kidney and kidney tumor segmentation in contrast-enhanced CT imaging: Results of the KiTS19 challenge," *Med. Image Anal.*, vol. 67, p. 101821, Jan. 2021, https://doi.org/10.1016/j.media.2020.101821.
25. J. Ma *et al.*, "AbdomenCT-1K: Is Abdominal Organ Segmentation a Solved Problem?," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.

44, no. 10, pp. 6695–6714, Oct. 2022, https://doi.org/10.1109/TPAMI.2021.3100536.
26. B. Rister, D. Yi, K. Shivakumar, T. Nobashi, and D. L. Rubin, "CT-ORG, a new dataset for multiple organ segmentation in computed tomography," *Sci. Data*, vol. 7, no. 1, p. 381, Nov. 2020, https://doi.org/10.1038/s41597-020-00715-8.
27. X. Li, P. S. Morgan, J. Ashburner, J. Smith, and C. Rorden, "The first step for neuroimaging data analysis: DICOM to NIfTI conversion," *J. Neurosci. Methods*, vol. 264, pp. 47–56, May 2016, https://doi.org/10.1016/j.jneumeth.2016.03.001.
28. J. R. Bumgarner and R. J. Nelson, "Open-source analysis and visualization of segmented vasculature datasets with VesselVio," *Cell Rep. Methods*, vol. 2, no. 4, p. 100189, Apr. 2022, https://doi.org/10.1016/j.crmeth.2022.100189.
29. C. Delong and S. Sharma, "Physiology, Peripheral Vascular Resistance," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2023. Accessed: Nov. 24, 2023. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK538308/
30. M. Ohishi, "Hypertension with diabetes mellitus: physiology and pathology," *Hypertens. Res.*, vol. 41, no. 6, Art. no. 6, Jun. 2018, https://doi.org/10.1038/s41440-018-0034-4.
31. N. Dib, F. Oberti, and P. Calès, "Current management of the complications of portal hypertension: variceal bleeding and ascites," *CMAJ*, vol. 174, no. 10, pp. 1433–1443, May 2006, https://doi.org/10.1503/cmaj.051700.
32. T. I. Oliver, B. Sharma, and S. John, "Portal Hypertension," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2023. Accessed: Nov. 24, 2023. [Online]. Available: http://www.ncbi.nlm.nih.gov/books/NBK507718/
33. N. Pfisterer, L. W. Unger, and T. Reiberger, "Clinical algorithms for the prevention of variceal bleeding and rebleeding in patients with liver cirrhosis," *World J. Hepatol.*, vol. 13, no. 7, pp. 731–746, Jul. 2021, https://doi.org/10.4254/wjh.v13.i7.731.
34. B. Hillen, B. A. H. Drinkenburg, H. W. Hoogstraten, and L. Post, "Analysis of flow and vascular resistance in a model of the cricle of Willis," *J. Biomech.*, vol. 21, no. 10, pp. 807–814, Jan. 1988, https://doi.org/10.1016/0021-9290(88)90013-9.
35. J. Mayet and A. Hughes, "Cardiac and vascular pathophysiology in hypertension," *Heart*, vol. 89, no. 9, pp. 1104–1109, Sep. 2003.
36. T. Fushiki, "Estimation of prediction error by using K-fold cross-validation," *Stat. Comput.*, vol. 21, no. 2, pp. 137–146, Apr. 2011, https://doi.org/10.1007/s11222-009-9153-8.
37. Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, 2020.
38. K. Hara, D. Saito, and H. Shouno, "Analysis of function of rectified linear unit used in deep learning," in *2015 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2015, pp. 1–8. https://doi.org/10.1109/IJCNN.2015.7280578.
39. Z. Zhang, "Improved Adam Optimizer for Deep Neural Networks," in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, Jun. 2018, pp. 1–2. https://doi.org/10.1109/IWQoS.2018.8624183.
40. A. Colin Cameron and F. A. G. Windmeijer, "An *R*-squared measure of goodness of fit for some common nonlinear regression models," *J. Econom.*, vol. 77, no. 2, pp. 329–342, Apr. 1997, https://doi.org/10.1016/S0304-4076(96)01818-0.
41. A. Bray *et al.*, "Pulse Physiology Engine: an Open-Source Software Platform for Computational Modeling of Human Medical Simulation," *SN Compr. Clin. Med.*, vol. 1, no. 5, pp. 362–377, May 2019, https://doi.org/10.1007/s42399-019-00053-w.