Modelling Affordance based on Physical Characteristics

Guanjie Liang

Abstract

As climbing continues to grow rapidly in popularity, climbers must develop a heightened awareness of their own physical capabilities and limitations. Unlike disciplines that rely on standardized motion patterns, climbing demands that climbers dynamically and quickly adjust their movement strategies to reach the goal as quickly as possible based on their individual physical characteristics such as height, arm span, and leg span, and requires continuous decisions about which holds are reachable by the hands or feet and how to progress towards a target while remaining balanced. We study this problem on multiple fixed walls whose annotations consist only of hold shapes. We estimate the center coordinates of the grab point by using the coordinates provided for each vertex, and we synthesize a start configuration by randomly selecting holds because no start holds are provided. A Graph Neural Network (GNN) is utilized to label every hold as unreachable (0), hand-only (1), foot-only (2), and both (3) by using geometric features derived from the current limb positions and normalized by the climber’s anthropometrics, such as height, ape index, arm span, and flexibility. On top of these predictions we compare two planners, the first one is a greedy baseline that moves hands and feet simultaneously toward the goal while keeping feet below hands and always select the furthest holds, and the other one is a posture-aware planner that moves one limb type at a time, prefers hands above feet, and encourages vertical alignment between hands and feet centers via an angle threshold and alignment cost.

We evaluate three different cohorts, which are casual, skilled, and elite, and each cohort is with 100 simulated climbers on the same wall, start, and goal. Using bar charts, we summarize completion rate, and average steps to goal for successful trials, including stratifications by key anthropometric factors. The results show that performance differences are driven primarily by individual body attributes, such as arm span, leg span, height, and ape index, rather than the cohort label alone. Climbers with larger effective reach complete more frequently and in fewer steps. The greedy planner achieves higher completion rates, whereas the posture-aware planner yields more vertically balanced sequences at the cost of success on some instances, which reveals a practical trade-off between progress and stability. We provide an evaluation protocol and analysis that can inform route setting and training feedback for different ability levels.

Ethics Statement

After discussing with my supervisor, we plan to use the dataset that contains the climbing route with hold points from [Kaggle](https://www.kaggle.com/datasets/tomasslama/indoor-climbing-gym-hold-segmentation/data) and simulate the climbers with different attributes in our research. Therefore, this project that do not require ethical approval at all, because it does not collect or process any relevant data.

Supporting Technologies

* Programming Environment: Code was developed using a recent version of Python 3 with Jupyter notebooks and plain scripts, utilizing data science libraries such as NumPy and Pandas.
* Deep Leaning Framework: PyTorch for model definition, training, and inference, tensors for moving CPU/GPU, and PyTorch Geometric for graph ops and batching.
* Version Control: Git with a GitHub repository was used to ensure that the project is securely stored and documented in the cloud.
* I used LaTeX through the online service Overleaf for the formatting of the thesis.

Acknowledgements

I would like to express my deepest appreciation to my family. Without their support and help, I would not have been able to obtain this master’s degree. They not only provided me with material assistance in my daily life, but also offered me tremendous emotional support. I would like to express my gratitude to my parents again.

I am extremely grateful to my Supervisor Dr. Sophie Baker. Her patience and professional guidance have undoubtedly been of great help to me. Whenever I encounter difficulties, she always patiently provides me with ideas and gives me plenty of room to freely express myself in the development of the project. And many thanks to all lecturers in this master programme. They showed me the way forward during the year and taught me lots of professional knowledge.

Chapter 1

Introduction

This chapter opens with background on indoor climbing and motivates climber-conditional reachability analysis as a tool for route setting and training. We then discuss the limitations of evaluations based on hierarchical or single-strategy assessments and introduce our approach. A graph neural network based on anthropometric data is combined with two planners, which are a greedy baseline planner and a posture-aware strategy planner. Finally, we highlight development challenges (centroid approximation, synthetic starts, and planner hyperparameters) and outline the research aims on ability-group differences, policy trade-offs, and sensitivity.

* 1. Project context and problem

Indoor climbing success depends jointly on wall geometry, holds affordances, and a climber’s body[1]. Route levels summarize difficulty for an average person, but for setters and coaches, they often need a more precise answer to who can do what, how they can do it, and in how many steps[2]. A computational pipeline is built for a fixed wall with annotated holds, a start and goal, and simulated climbers described by anthropometrics, which is to predict whether the goal is reachable, and how many steps it will take under a chosen movement strategy.

We represent the wall as hold graph whose nodes hold centroids that are derived from the provided shapes and whose edges connect spatial neighbors. A reachability model is built by graph neural network, labels each hold as unreachable, hand-only, foot-only, or both, which is inspired by Nishad[3]. On top of these predictions, we generate motion sequences using two planners. One is a greedy baseline that moves hands and feet simultaneously toward the goal and keeps feet below hands. The other one is a posture-aware planner that moves hands or feet at a time, and encourages vertical alignment between hand and foot centers through an angle threshold and alignment cost. For evaluation, we simulate three cohorts (casual, skilled, elite) of 100 climbers each, sampled over realistic anthropometric ranges[5][6][7], such as height, ape index, weight, and strength.

* 1. Practical and Methodological Implications

Simulating reachability rather than relying on rough grades brings two immediate benefits, which are route setting and training feedback. For route setting, setters can test whether a proposed route is unfairly biased towards a specific body type, especially the climbers who are taller and have longer arm spans[2][8], and identify where to add intermediate footholds or handholds to reduce the impact of physical differences[4]. For training feedback, coaches and climbers receive targeted guidance. For instance, when performance is limited by poor foot follow-through rather than hand reach[1], or when movement patterns consistently violate upright posture and cause instability[9]. Coaches and climbers can use this feedback to tailor training, such as building strength or improving hold point selection.

The broader significance of this work lies in its methodological contribution. We present a lightweight and transparent tool that integrates two complementary components: a learned model of local hold affordances, which is implemented as a graph neural network, and a set of explicit motion rules encoded in path planning algorithms. This design allows performance results to be clearly traced back to the decision logic of the GNN’s predictions or planner, e.g. whether the goal can be reached by climber or the number of moves required. This traceability not only facilitates targeted optimization of models or rules, but also enhances the system's interpretability and reliability for setters, coaches, and researchers.

* 1. Prior work and selected approach

The classic difficulty assessment method is a combination of expert judgement and coarse statistical data, such as height and grade[8]. Recent research on accessibility and posture planning has prompted us to learn about the reachability of each grip point based on the climber’s condition, and then plan based on these predictions[1][12]. Our approach follows the two-stage design for transparency and speed: Climber-conditioned reachability and Policy-level simulation.

For Climber-conditioned reachability, the GNN predicts unreachable, hand-only, foot-only, and both labels for each hold by mixing features of the distance between the current position and the current hands/feet position, and normalizes them based on the range of arm and leg reach. Anthropometric vectors, such as height, ape index, spans, flexibility, and leg-length factor, modulate the node embeddings. Thus, the same wall will produce different reach maps for different climbers[3][11].

For Policy-level simulation, we apply two simple and interpretable planners, rather than solving full-body dynamics. The first planner is greedy progress, which can achieve the fast expansion of the reachable frontier[4]. The second planner is posture-aware, which achieves softer progress but higher likelihood of hands-over-feet and near-vertical alignment[12].

This kind of separated design allows us to study trade-off between progress and stability and attribute failure to limitations in model coverage or conservative movement choices by planners.

* 1. Central Challenges
* Sparse supervision and geometry: The dataset from Kaggle contains hold shapes (the coordinates of each vertex) but no official start configuration. We approximate each hold by its centroid and generate a synthetic start, which may miss directionality/friction effects of real holds.
* Coupling perception and policy: A model will mark many holds reachable based on provided position, but a conservative strategy may result in a deadlock. Conversely, an aggressive strategy may succeed without strict posture control. It is hard to clarify these relationships.
* Fair comparisons across abilities: To ensure differences reflect body attributes and policy rather than random initialization, we fix the walls and goals and evaluate all cohorts on the same set of initial states. In contrast, randomized positions of climbers were employed in training to produce labels and broaden coverage.
* Hyperparameter sensitivity: New knobs are introduced in posture-aware planning, such as angle threshold, and alignment weights, which can shift the balance between success and stability.
  1. Research Aims
* Learn hold affordances conditioned on the climber: Train a GNN that tags each hold as unreachable, hand-only, foot-only, or both by using current position and climber’s attributes.
* Compare two movement strategies on the same predictions: Implement a greedy policy, which moves both hands and feet simultaneously and always selects the closest reachable hold to the goal. In addition, a posture-aware policy also needs to be implemented, which moves only one limb type at a time, considers the body’s balance when attempting to move to the closest reachable hold to the goal, such as keeping the body as vertical as possible, and also considers whether the four hold points are reasonable.
* Establish a reproducible evaluation protocol across abilities: Simulate casual, skilled, and elite cohorts (100 each) on a single wall with same start and goal to record the completion rate and the number of steps to goal (success: number of steps, failure: -1).

Chapter 2

Background

This chapter establishes the technical background for our study and positions it within prior work. We first review core climbing biomechanics, such as height, ape index, and flexibility, and how to set hand/foot reach and stable posture, also including a standing-reach that we later use for over-stretch guarding. We then model a wall as a graph of holds, explain our state-dependent node features that are recomputed after each move, and motivate casting per-hold affordances as a four-class node classification problem with a GNN. Building on these predictions, we contrast two lightweight planners: greedy and posture-aware. We then discuss assumptions about static and dynamic reach, and describe our synthetic supervision pipeline. The chapter closes by identifying research gaps and explains the basis for our evaluation protocol and design choices.

2.1 Climbing Biomechanics

Climbing movement is fundamentally geometric. Thus, feasible actions depend on where the wall’s holds lie relative to the climber’s body[16]. In our experiments, we assigned six anthropometric factors to all climbers, which are height, weight, ape index, strength, flexibility, and leg length factor. In practice, we only used height, ape index, flexibility, and leg length factor. Height and arm span (height × ape index) set the range for hand reach[11], flexibility and leg span (height × leg length factor) expand foot reach and high-step range[14], and ape index summarizes upper-limb advantage.

We use an analytic reach oracle to test whether a hold is reachable by hands or feet given the current contacts[11]. On top of these predictions, we evaluate two planners – greedy baseline and posture-aware planner. For the greedy planner, it moves hands and feet simultaneously to reachable holds closest to the goal in a step, enforce feet below hands, and there are no posture cues, which accelerates progress but can produce stretched poses[4]. For the posture-aware planner, it will consider the positions of hands and feet, which will keep feet below hands as much as possible[9]. In addition, it can monitor the angle to vertical between hand and foot centers[14], and it can guard against over-stretch by checking whether vertical hand-foot separation exceeds a given threshold of standing reach. When this “reach guard” triggers, we move whichever limb center is farther from the goal to bring the body back to a reasonable range[13]. These heuristics are simple, interpretable surrogates for fuller balance models and match common coaching cues[13][15].

2.2 Modeling a wall as a graph of holds

Each wall is represented as a graph, whose nodes are holds (we use hold centroids parsed from region shape attribute) and whose edges connect spatial neighbors, e.g. KNN. This supports local message passing and keeps the model invariant to permutations of holds[17].

Node features in our implementation are state dependent and get recomputed after every simulated move:

* Centroids of hold: (x, y)
* Distances from the node to the current hands and feet: both the mean and minimum, each normalized by the climber’s arm reach or leg reach.
* Binary flags: marking whether the node is currently used by a hand or a foot.

This encoding exposes the GNN to the instantaneous affordances of the state by indicating the position of new support points relative to the currently established support base without explicit dynamic information[3].

2.3 Per-hold affordances with GNN

We cast reachability as node-wise affordance classification with four classes, which are unreachable, hand-only, foot-only, and both (0, 1, 2, 3 in the logits’ argmax). The model performs message passing over the hold graph and outputs a logit vector per node[19].

Supervision via synthetic starts. Rather than relying on a single official start, we generate (K=20) randomized initial limb configurations per climber-wall pair under simple constraints[14]. For each sampled start and each set of anthropometrics, we label every hold with the reach function. This random augmentation method generates a large and diverse training set, which enables GNN to encounter a wide range of relationships between bodies and walls[20]. In our typical scale before filters: 100 climbers × 15 walls × 20 starts = 30000 graphs per cohorts.

2.4 Lightweight planners on top of the GNN

We evaluate two simple planners that generate a series of interaction sequences based on the classes predicted by GNN. These two planners reevaluate the graph features and rerun GNN after each move[19][21].

2.4.1 Greedy planner (progress-oriented)

* In each step, this planner will pick two hands and two feet among nodes predicted reachable, prioritizing closest-to-goal choices while keeping feet below hands[4].
* Hands and feet move simultaneously in a step.
* Tends to maximize upward progress, often high completion, but can be stretched, and less stable postures[14].

2.4.2 Posture-aware planner (stability-oriented)

* Move one limb type at a time: if the hand-foot center line deviates from vertical beyond a threshold, move feet to realign under the hands, otherwise move hands[12].
* When moving hands: prefer candidates above the feet (with a small vertical margin) and choose a pair that is close to the goal.
* When moving feet: prefer candidates below the hands, exclude occupied hand holds, and choose the pair that can minimize a center alignment cost plus a goal proximity term[22].
* Standing-reach guard: if vertical hand-foot separation is too large, override the choice and move the limb type which is farther from the goal first.
* This policy explicitly encodes uprightness and “hands-above-feet” structure, typically yielding more balanced sequences at a potential cost in success[13].

Based on these two planners, we also provide the non-visual calculation to return the number of steps to success or -1 on failure, which enables large batch evaluation.

2.5 Affordances in climbing

Affordance theory understands perception from the perspective of the opportunities for action provided by actors and their environment. In climbing, a hold affordance refers to a point that climbers can grip or step on relative to their own bodies and the current contact point. This frame justifies predicting per hold affordances (unreachable, hand-only, foot-only, and both), the affordances of all other holds will change by using state-dependent node features when the base of support changes. It also explains why simple geometric cues, such as feet-below-hands, and vertical alignment, can approximate more complex balance models[23].

2.6 Static and dynamic reach

Sport science distinguishes static reach (static posture, no swing) from dynamic reach (momentum). Our oracle and planners target the static and semi-static regime typical of controlled modern moves. This choice can produce consistent labels and aligns with graph-based reasoning. However, it could underestimate explosive jumps and sequences that require high coordination, so that this is a known limitation and a direction for future research[16].

2.7 Anthropometry and standing-reach normalization

Height and arm span set first-order upper-limb reach, ape index summarizes their ratio, flexibility (hip and ankle flexibility) and leg span govern high-steps and body elevation. A simple standing-reach approximation from dataset on anthropometric measurements[26]:

Supports body-scaled normalization and our 75% guard against over-stretch. Normalizing distances by arm and leg reach makes features more comparable across climbers and reduces fake correlations related to absolute pixel scales[26].

2.8 Label generation with synthetic starts

In the absence of official starts and measured athletes, we adopt a fully synthetic supervision strategy: walls are abstracted by hold centroids, climber anthropometrics are sampled from plausible ranges, and an analytic reach oracle provides per-hold affordance labels under randomized initial states. This yields broad coverage of body–wall–state combinations and motivates our per-node, state-dependent formulation[25].

2.9 Research Gap

Although the popularity of rock climbing has attracted a large number of researchers to conduct calculations and analyses, several gaps remain:

* Lack of per-hold, state-dependent, individualized affordances: Prior work typically predicts route difficulty or aggregate success for an average climber, or studies generic affordances, but does not model reachability at the hold level as a function of the current positions and explicit anthropometrics. To our knowledge, there are currently no studies using GNN to output node-level labels that simultaneously satisfy state conditions and climber conditions.
* Limited, interpretable posture modeling in planners: Most planning baselines are greedy toward the goal or rely on untransparent policies with little explicit treatment of posture and stability. We did not find a systematic evaluation of simple, interpretable geometric cues as planning constraints layers over learned affordances.
* Spare supervision strategies for state space: Existing datasets barely provide dense state labels across many starts. There is a gap in scalable supervision, which utilizes an analytical reachability oracle to label each hold state for each climber-wall pair across multiple random initial states, so that the model can learn how reachability changes as the support set changes.
* Cohort-level evaluation that cleanly separates ability from initialization: Comparative studies across ability levels are rare and often confounded by different starts. It is imperative to establish a protocol for fixed walls and goals, apply multiple initial conditions to each climber-wall pair, and summarize the results to ensure that the observed differences reflect physical characteristics and policy influences rather than just initial condition specificity.
* Outcome metrics beyond completion only: Many works report success/failures but omit the distribution of the number of steps to goal, which is crucial for training feedback and route setting.
* Reproducible, lightweight pipelines: Prior studies often lack open, modular setups where one can adjust angle thresholds, vertical margins, and alignment weights. A simple, reproducible pipeline is missing for broad cohort and hyperparameter-level analysis.

Chapter 3

Methods

3.1 Data & Preprocessing

In our project, the data we used contains climbing walls, climbers with anthropometrics, and initial states. The climbing walls are sourced from Kaggle, while the climbers with anthropometrics and initial states are simulated by us.

3.1.1 Climbing Walls

The initial data for the climbing wall consists of multiple hold points, with each row representing a hold point. A hold point is a polygon composed of multiple vertices, which are represented by x and y coordinates. We first remove empty walls (walls with zero valid holds) to avoid noisy, degenerate examples and reduce dataset size. Then, we consolidate the holds belonging to the same wall into a single record (one row per wall), which shortens the dataset while increasing unit depth (all holds of a wall are stored together). This structure simplifies downstream batching and makes us compute wall-level diagnostics once. In addition, we build a shapely.geometry.Polygon and extract interpretable 2D descriptors for each hold polygon, which are area, perimeter, aspect ratio, and circularity. These features further enhance node attributes by adding simple, easy-to-understand geometric shapes that are associated with holding. What’s more, we compute the centroid (x, y) for each hold, yielding the wall’s set of hold centers:

3.1.2 Climbers

At the beginning of the project, I reviewed a lot of literature and browsed through many climber data from rock climbing projects, but I could not find any suitable climber data with anthropometrics. So, for the climbers’ data, we used simulated data, which was generated by Python simulations. We divide climbers into three different skill levels: casual, skilled, and elite. For each climber in each level, we sample several attributes based on different ranges to each climber[5][7].

Table 1: Simulated climber attributes by skill level

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Climbers | Height | Ape Index | Weight | Strength | Flexibility | Leg Length Factor |
| Casual | 165±5 | 1.00±0.03 | 70.5±14.78 | 70±10 | 4-6 | 0.5±0.05 |
| Skilled | 170±5 | 1.02±0.02 | 69.9±10.24 | 85±10 | 6-8 | 0.51±0.06 |
| Elite | 177±8 | 1.05±0.03 | 67.2±13.6 | 105±10 | 8-10 | 0.53±0.05 |

From the character attributes in Table 1, we also extend the arm span (Height \* Ape Index) and leg span (Height \* leg length factor). Arm span determines the effective reach envelope of the upper limbs, directly constraining the set of holds that can be contacted with the hands[11]. In theory, climbers with a longer arm span relative to their height (higher ape index) are usually able to bypass the middle foothold and utilize a wider grip position. In contrast, leg span governs the maximum high-step and the ability to reposition the lower body upward. A greater leg span and leg length factor allow climbers to stabilize postures by maintaining feet below hands and to reduce over-stretch during upward progression[27]. These factors collectively constitute the core anthropometric determinants of individual reachability, which are the essential predictive indicators in our modelling framework.

3.1.3 Initial States (Starts)

In addition to the information of climbers and walls, we also simulated the position of climbers on each wall for subsequent analysis training. The method we use to generate starts is radius domain sampling: given the centroids of all hold points on a wall and the body measurements of any climber, we randomly generate a set of initial hands and feet contact points that satisfy geometric feasibility constraints, such as “hands above feet”. In a word, it will randomly select a hold point as the center and randomly select four hold points within a circle with a certain physical radius with that point as the center. Then, it will assign them as hands and feet automatically based on their y-values. If the constraints are not met, retry. If the neighborhood is too less to select four points, gradually expand the radius and try again.

For hyperparameter and constraints, we set the initial sampling radius to with k=0.6, and progressively expand it by [1.0, 1.25, 1.5, 1.75, 2.0] while capping . For each radius we draw up to max\_trials=50 centers. Feasibility is enforced by hands-above-feet and within-limb span[23]. Figure 1 is the heatmaps of initial states divided by all starts, feet, and hands.

图表

AI 生成的内容可能不正确。图表

AI 生成的内容可能不正确。图表

AI 生成的内容可能不正确。

Figure 1. Starts Heatmaps: All / Feet / Hands

Start center heatmaps correlate strongly with hold density because our radius sampler seeds from existing holds. Hand and foot center heatmaps exhibit the expected vertical offset induced by the ‘feet-below-hands’ constraint. We also validated the sampler by inspecting the distribution of vertical hand-foot separations and by comparing per-wall center heatmaps to raw hold density, which both confirm the intended behavior.

3.1.4 Merge Climbers, Walls, and Starts

After the first three steps, we now have all the data needed for the experiment, which are the climbing walls (the centroids of each hold point and its features), the climbers (including the physical characteristics of each climber and the level), and the initial starts (obtained by radius sampling). Each record stores like figure 2:

* The wall graph (hold centroids and geometric features if needed).
* The climber dictionary (height, ape index, flexibility, strength, weight, and leg length factor).
* The hands hold points: .
* The feet hold points: .

日历

AI 生成的内容可能不正确。

Figure 2 is the visualization of climbing wall hold-contact.

The gray circles represent all climbing holds on the wall.

The red squares represent the holds currently occupied by the climber’s hands, and the purple squares represent the holds currently occupied by climber’s feet.

Figure 2. Visualization of one sample

3.1.5 Per-hold Labels (Affordances)

* Labels: For each node/hold on wall under start and climber , we assign

* Reach Radius: From the climber’s anthropometrics, we compute single-arm and single-leg static reach radius via get\_reach\_ranges() function. Distances are measured in centimeters using pixel\_dist\_to\_cm().
* Decision Rule: If , then set hand\_ok True, else set false. It is the same rule for feet. We do not distinguish left/right sides, and multiple nodes may share the same label.
* State dependence: Because labels depend on the current contacts (hands and feet hold positions) and climber’s features, the same wall receives different labels across starts and climbers.
* Degenerate-case filter: To maintain effective supervision and improve the validity of data, we discard samples with .
* Limitations: Labels capture static/semi-static geometric reach only, which ignores momentum and contact quality. Thus, we note it as future work.

3.2 Graph representation for the GNN

For each sample , a wall, a climber, one start (two hands and two feet), and a node-wise label vector are built as a hold graph in figure 3. We follow PyTorch Geometric’s Data to implement the graph, which consists of node matrix, edge index, graph-level climber stats, and node labels y.

图示

AI 生成的内容可能不正确。

Nodes are hold centroids; edges connect spatial neighbors. Colors denote per-node affordances for the current state: grey = unreachable, blue = hand-only, orange = foot-only, and green = both reachable. This illustrates both the sparse local connectivity used by GNN and the state-dependent nature of labels.

Figure 3. KNN hold graph for one wall and start

3.2.1 Node

Each node corresponds to a hold centroid. The node feature vector concatenates static geometry and state-dependent affordance cues. For node , we compute distances to current contacts and normalize them by the climber’s reach ranges. Node feature order exactly as in code:

* x – hold x
* y – hold y
* mean\_d\_hand / arm\_reach – mean distance to the hands, clipped to [0, 3]
* min\_d\_hand / arm\_reach – min distance to the hands, clipped to [0, 3]
* mean\_d\_foot / leg\_reach – mean distance to the feet, clipped to [0, 3]
* min\_d\_foot / leg\_reach – min distance to the feet, clipped to [0, 3]
* is\_hand\_point – 1 if the node is within 1.0 cm of any current hand contact, else 0
* is\_foot\_point – 1 if the node is within 1.0 cm of any current foot contact, else 0

Clipping is the process of limiting a value to a range between a minimum and a maximum value, which is also used a lot in node features. When two points are far apart, a large value is generated; when two points are close, a small value is generated. When these two values are placed side by side, the difference becomes more obvious, which is unfavorable to the model training. Therefore, it is important to use clipping, which can improve numerical stability and prevents a few far-away holds from dominating the scale[19][29]. And it can help the model to focus learning on the physically relevant regime (about 3 times reach), and yield smoother gradients for the GNN[17].

3.2.2 Labels

Provided externally to the builder and stored as 0 = unreachable, 1 = hand-only, 2= foot-only, and 3 = both. Labels are state-dependent, which depends on hands, feet, climber’s reach, and current position, and are recomputed for each start.

3.2.3 Edges

Rather than building a fully connected graph, we construct an undirected KNN graph on the centroids. It fits NearestNeighbors(n\_neighbors = min(10, N)) on N centroids. For each node, we add edges (i, j) for its neighbors j, and exclude self-loops. Last, we store each edge once as a sorted pair in a Python set to remove duplicates. Here are the benefits of KNN compared to fully connected graphs:

* Lower computation/memory: Fully connected has , and KNN has . For tens-hundreds of holds, KNN reduces forward/backward cost and GPU memory by an order of magnitude.
* Mitigates over-smoothing: Fully connected floods each node with near-global information in a single layer, risking feature averaging. However, sparse KNN neighborhood can maintain cross-layer distinguishability.
* Noise robustness: Fully connected propagates outliers globally, but KNN can confine the influence of noisy/odd holds to local regions.
* Geometric fidelity: KNN retain local topology such as clusters that fully connected would wash out.

3.2.4 Climber conditioning

We attach a graph-level tensor with six anthropometric features (as floats):

* Height
* Ape index
* Flexibility
* Leg length factor
* Arm span
* Leg span

These same quantities are also used inside feature construction via hand reach and foot reach.

3.2.5 Current position on the graph

Current position is important for reachability, which depends on the distance between holds. For convenience during simulation, we also store the actual position coordinates:

* Hands:
* Feet:

Planners update these arrays step-by-step, and after that, node features will be recomputed and update the values to each feature.

3.3 Model: per-hold affordance predictor

We cast reachability as node-wise 4-way classification (unreachable, hand-only, foot-only, and both). The graph is a PyG Data object with node features, a KNN edge index, a graph-level climber vector, and current position coordinates.

3.3.1 Architecture

The overall architecture of Reachability GNN follows a multi-branch design in Figure 4. The backbone processes graph-structured node features through stacked Graph Attention Network (GAT) layers to capture spatial dependencies among holds. In parallel, the anthropometric features are projected into the same latent space by a climber embedding branch, which allows personalization of predictions. A dedicated flag head separately encodes binary indicators of current hand and foot assignments, which is served as an inductive bias for the affordance classification task. Finally, the output from the main branch and the flag head are combined to produce per-node logits over the four affordance classes.

图示

AI 生成的内容可能不正确。

Figure 4. The Architecture of ReachabilityGNN

Building on this overview, the following subsections describe each architectural component in detail:

* Backbone: The backbone is to turn raw node features into context-aware embeddings by attending to spatial neighbors, which consists of two layers built by GATConv and ReLU. This setup follows the attention-based message passing framework of graph neural networks[30], which produces node embeddings that encode local geometry and connectivity while remaining invariant to node order[31].
* Climber embedding: The climber embedding can personalize predictions, so the same wall yields different reach maps for different climbers. In addition, it can use the batch vector to attach the corresponding climber embedding to every node in that graph, and it provides a graph-level context aligned to each node, which enables the classifier to modulate decisions by body features[32][33].
* Flag head: The flag head injects a simple and interpretable bias from “is this node currently occupied by hand/foot?” without going through message passing, and learns priors to improve calibration in states where occupancy matters[34].
* Output fusion: It combines context-aware node embeddings with climber conditioning and the flag prior into per-node class logits, which improves robustness without complicating message passing[35].

3.3.2 Inputs

The input of the model is a graph that consists of nodes, edges, graph-level climber tensor, labels, and current contacts. The details are in section 3.2.

3.4 Objective & optimization

A weighted focal loss is used to train on per-node logits, which cope with the heavy class imbalance (most of them are unreachable):

Class weights w is computed from the training labels using balanced method, and then clipped to [0.5, 5.0], normalized by their mean, and clipped again to avoid extreme values before converting to a tensor. After all these processing, we pass weights into focal loss. For optimizer and schedule, Adam with a StepLR scheduler is used in model training. Batches are formed with PyG’s Batch (graphs are concatenated, and each graph keeps its own climber vector via the batch index), and we shuffle graphs each epoch. This combination can bring many benefits:

* Handles class imbalance: Weighted focal loss de-emphasizes abundant, easy negatives (class 0) and focuses learning on positives (class 1 - 3)[36].
* Stable weighting: Clipping and mean-normalizing the weights prevents exploding gradients from extreme priors and improves convergence stability[37].
* Learning-rate schedule: StepLR halves the learning rate periodically, helping escape early plateaus while reducing overfitting late in training[38].
* Shuffled graph batches: Shuffling improves generalization, and PyG batching preserves per-graph climber conditioning correctly[39].

3.5 Planners layered over the GNN

We evaluate two algorithms that repeatedly: (i) run the GNN, (ii) pick next hold points based on predicted classes and simple geometry, (iii) record the new hands and feet coordinates back into the graph, and (iv) recompute all node features before the next step. When the goal node’s predicted label becomes non-zero, it means success.

3.5.1 Greedy (progress-oriented)

* Hand Selection: At each step, the planner first selects two hand placements from the set of holds predicted as either hand or both. The first hand hold is chosen to be the one closest to the goal, and then the second point is chosen to be the one closest to the first hand hold, which ensures that the two hands remain reasonably coordinated.
* Foot Selection: The planner selects two footholds from the set of holds predicted as either foot or both, excluding any that have already been chosen by hand. Among these candidates, the planner will prioritize the holds underneath and then take the two closest holds to the goal.
* Update and Iteration: Hands and feet are moved simultaneously in each step. After every move, the node features are recomputed to reflect the new body configuration, and the GNN is re-evaluated to update the set of reachable holds. This process is repeated until the goal is reached or there are no valid moves.

3.5.2 Posture-Aware (Stability-oriented)

* Stepwise movement: The planner moves only one limb type at a time. At each step, it will calculate the line connecting the center of current hands and feet. If the angle of this line relative to the vertical exceeds a predefined threshold, the planner moves the foot to realign it below the hand. Otherwise, it moves the hands, giving preference to candidates that lie above the feet.
* Standing-reach guard: To prevent over-stretch, the planner incorporates a reach constraint. Standing reach is estimated from the climber’s height and arm span as

.

If the vertical separation between hands and feet exceeds 75% of this value, the planner overrides the default choice and moves whichever center is farther from the goal.

* Foot Selection: When moving the feet, the planner excludes holds already occupied by the hands and prioritizes footholds located below the hands. Among these candidates, it will combine lateral alignment and proximity to the target to select the pair that minimizes the cost function.
* Hand Selection: When moving the hands, the selection follows a rule that choosing the closest to the goal then choosing the nearest neighbor, restricted to holds above the feet.

3.6 Metrics

Model evaluation is aligned with our research goals. We assess what the planner achieves on the wall and why the GNN’s predictions are credible. At the planner level, we report completion rate and steps-to-goal for successes. At the node level, we quantify predictive quality with per-class precision/recall/F1/support over the four affordance labels, and provide true and predicted values traces.

3.6.1 Node-level metrics

* Per-class: On the held-out val graphs, we evaluate node-wise predictions for the four classes {0: unreachable, 1: hand-only, 2: foot-only, 3: both} using sklearn.metrics.classification\_report.
* True vs. predicted traces: For qualitative inspection, we plot ground-truth and predicted labels along node index for exemplar graphs.

3.6.2 Planner-level metrics

* Completion rate: For each cohort (casual, skilled, and elite) and wall, we run all randomized starts and report the fraction that reach the goal. Failures are counted in the denominator.
* Steps-to-goal: For successful runs only, we record the number of steps and summarize with means. We show the distribution by cohort in the form of grouped histograms, and the legend annotates the sample size n and the success rate of the cohort.

3.6.3 Reproducibility notes

Evaluation uses the final trained weights with no augmentation. Success is defined by the model predicting the goal hold as reachable (1,2, or 3), and failures are assigned step count as -1, which will be excluded from the steps histogram. Hyperparameters, such as alignment wights, are fixed as specified in the configuration.

Chapter 4

Results

This chapter reports the results of the experiments carried out on the Reachability GNN model, which is divided into Experimental Snapshot, Node-Level Performance, Reachability Maps under a Fixed Wall & Start, Planner Outcomes, and Ablations on Posture-Aware Policy.

4.1 Experimental Snapshot

Table 2 is the summary of our experiment. For each wall and climber, we generated 20 randomized starts using radius sampling, yielding casual, skilled, elite climbers per cohort. In total, we produce N = 15×20×300 = 90000 labeled hold-graphs, and after filtering, there are 54655 graphs remained for training and evaluation.

Table 2. Summary of Experiment

|  |  |  |  |
| --- | --- | --- | --- |
| Category | Item | Value | Notes |
| Data | Walls | 15 | After filtering empty/degenerate walls |
| Holds per wall | Different walls have different numbers of hold | Centroids from polygon parsing |
| Starts per wall/climber | 20 | Radius sampling, feasibility constraints |
| Cohorts (casual, skilled, elite) | 100 / 100 / 100 | Each class has its own ranges of features |
| Total Climbers | 300 | Sum across cohorts |
| Total graphs | 54655 | The ideal number of graphs should be 15×20×300=90000. However, after checking for duplication and filtering, only 54655 remained. |
| Split | Train / Val / Test | 70% / 10% / 20% | At graph level |
| Model | Backbone | ReachabilityGNN | GATConv → concat climber  Embed → MLP + flag head |
| Node features | 8-D | (x, y), 4 clipped distances, 2 flags |
| Climber features | 6-D | Height, ape index, flexibility, leg length factor, arm span, leg span |
| Optimization | Loss | Focal (gamma=2.0) | Down-weights easy class-0 negatives |
| Optimizer | Adam | lr 1e-3 |
| Batch | PyG Batch | Multiple graphs per step |

In figure 5, the training loss decreases steadily across epochs, with sharper improvements immediately after each LR drop. The slope flattens after around 40 epochs, which indicates diminishing returns. The schedule steps align with visible loss kinks (approximately 15/30/44) correspond to visible loss kinks, which confirms the effectiveness of linear regression decay. There is no instability spikes observed, and the minimum training loss occurs at epoch 48. For Val macro-F1, it steadily improves from around 0.6 in the first few epochs to 0.979 at epoch 48. In a word, figure 5 confirms that ReachabilityGNN trains smoothly under focal loss, and the class-imbalance is handled effectively. The step LR schedule plays a key role in extracting the final performance gains, which achieves near-ceiling validation set performance ultimately.

图表, 折线图

AI 生成的内容可能不正确。

Figure 5. Training Dynamics (Reachability GNN)

4.2 Node-Level Performance

We evaluate the GNN at the node level (per hold) on the test data. Metrics are computed over all nodes in the evaluation graphs, using macro-F1 to balance class imbalance and per-class precision/recall/F1/support for interpretability.

4.2.1 Overall Classification Quality

Table 3 is the overall classification quality on testing set. It is obvious that the class distribution is highly imbalanced, with unreachable 83.5%, both 10.7%, hand 4.4%, and foot 1.5%. However, the model reach a high performance with macro-F1 0.98 for the 4-way node classification task. For unreachable, it shows near-performance on the dominant class. For hand, precision is slightly lower than recall, consistent with a few false positives near the hand-reach boundary. For foot, it is same as hand, which have a very high recall with a small precision drop at the threshold. For both, it performs a high precision with occasional misses to a single-limb class when one limb is marginally farther.

Table 3. Per-class precision, recall, F1, and support

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
| Unreachable | 1.00 | 1.00 | 1.00 | 1106486 |
| Hand-only | 0.93 | 1.00 | 0.96 | 58441 |
| Foot-only | 0.94 | 1.00 | 0.97 | 19201 |
| Both | 1.00 | 0.98 | 0.99 | 141580 |
|  |  |  |  |  |
| Accuracy |  |  | 1.00 | 1325708 |
| Macro avg | 0.97 | 0.99 | 0.98 | 1325708 |
| Weighted avg | 1.00 | 1.00 | 1.00 | 1325708 |

4.2.2 Qualitative Alignment

Figure 6 shows four representative graphs where we plot ground truth node labels (solid green) and model predictions (dashed red) against node index. From left to right and top to bottom, we name these four figures large graph, medium-A graph, medium-B graph, and small graph, respectively. For the large graph with around 250 nodes, a long zero baseline is followed by a dense block of reachable nodes and two curves perfectly overlap. For the medium-A graph with around 120 nodes, two curves again perfectly overlap, which confirms that the performance of the model is good. For the medium-B with around 75 nodes, the curves near-perfectly match except for a single one-node blip at the boundary (0 and 1), which is consistent with the boundary condition approaching the threshold. For the small graph, a compact reachable segment around 32 to 42 is recovered cleanly. And we observe a brief mismatch between 0 and 1.

Figure 6. True vs. Predicted Per-Node Affordances

图表, 条形图, 直方图

AI 生成的内容可能不正确。

图表, 直方图

AI 生成的内容可能不正确。

图表, 直方图

AI 生成的内容可能不正确。

图表, 折线图

AI 生成的内容可能不正确。

It is found that almost all deviations are hand-both or foot-both within 1-2 nodes, rather than gross misclassifications, which mirrors the class report. The few mutations of 0 and {1, 2} occur at the spatial edge of reach, which is expected because of the use of discrete retention, a 1 cm “contact”, and a distance that was clipped and range normalized. Once a region is clearly within reach, the prediction results become stable and reliably match “both”, which indicates that the flag head and climbing conditions help eliminate ambiguity in movement.

4.3 Reachability Maps under a Fixed Wall & Start

To demonstrate that the model learned the relationship between climber’s anthropometrics and per-hold affordances, we created Figure 7. In figure 7, we visualized 4 climbers with fixed wall and start. Gray dots represent unreachable holds; blue dots represent the holds that are only reachable by hand; yellow dots represent the holds that are only reachable by foot; green dots represent the holds that are reachable by both hand and foot; red boxes represent the current holds for hands; purple boxes represent the current holds for feet; and red-bordered circles represent the goal hold. The four pictures correspond to the four climbers from different cohorts, and their anthropometrics can be checked in table 4.

图表, 散点图

AI 生成的内容可能不正确。 图表, 散点图

AI 生成的内容可能不正确。 图表, 散点图

AI 生成的内容可能不正确。 图表, 散点图

AI 生成的内容可能不正确。

Figure 7. Fixed Wall and Start with different climbers

Table 4. Climbers’ anthropometrics in figure 7

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Height | Ape Index | Flexibility | Leg Length Factor | Arm Span | Leg Span |
| Climber 1 | 163.3 | 1.0 | 5.0 | 0.510348 | 163.3 | 83.3398 |
| Climber 2 | 171.6 | 1.02 | 6.0 | 0.601341 | 175.032 | 103.19011 |
| Climber 3 | 171.6 | 0.99 | 6.0 | 0.51023 | 169.9434 | 87.55547 |
| Climber 4 | 185.3 | 1.05 | 8.0 | 0.513411 | 194.565 | 95.1351 |

4.3.1 Climber-1 vs Climber-4

* Climber-1 exhibits the smallest “hand ring” (blue) and a thin “both” band (green) just above the feet. The foot region (orange) is compact, which confines to the lower-left cluster of holds near the start.
* Climber-4 shows a significant upward and lateral expansion of the blue area, and a noticeable thickening of the green band where the hands and feet overlap. The orange area also expands, but the extent of the expansion is not as significant as that of the blue area.

Arm span and height inflate the hand-reachable set, and higher flexibility and longer leg densifies the “both” belts

4.3.2 Climber-2 vs Climber-3

* Both climber 2 and climber 3 are 171.6 cm tall, but climber 2 has longer arms and, crucially, a much larger leg-length factor, with slightly higher flexibility. As a result, the orange area expanded significantly, the green band became wider, and the blue area also expanded moderately.
* Climber-3 presents a significantly reduced orange area and a thinner green overlap, despite being the same height.

After the comparison, we found that holding height constant isolates the effect of limb proportions. Leg span dominates foot reach and the overlap, while ape index slightly adjusts the range of hand extension.

4.3.3 Across all four panels

Across the four reachability maps, consistent patterns emerge. Hand-reachable mass increases in the order climber-1 < climber-3 < climber-2 < climber-4, tracking arm span and height. Therefore, climber-4 has the largest range of hand options, while climber-1 has the smallest. Foot-reachable mass follows climber-3 ≲ climber-1 < climber-4 < climber-2, aligning with leg span and flexibility. Thus, climber-2 enjoys the largest foothold envelope, and climber-3 gets the smallest. The both-reachable band widens as leg span/flexibility increases, roughly following climber-1 ≲ climber-3 < climber-4 ≲ climber-2, because a wider stance overlaps more with the hand range. Overall, the number of reachable holds for the hand is significantly positively correlated with arm span, the number of reachable holds for the foot is significantly positively correlated with leg span and flexibility, and the reachable bandwidth for both is most closely related to the interaction between leg span and flexibility.

4.4 Planner Outcomes

We build two different planners, greedy and posture-aware, and compare them based on different cohorts. Two different climbing walls are used to compare the two planners, and figure 8 shows a visualization of these two walls, figure 9 shows the full sequence of hand and foot moves to the goal by using greedy planner, and figure 10 shows the combination of walls and planners. The same row shows the results for the same climbing wall, and the same column shows the results for the same planner.

图表, 散点图

AI 生成的内容可能不正确。 图表, 散点图

AI 生成的内容可能不正确。

Figure 8. Visualization of two walls in planners’ comparison (Left: Wall-1, Right: Wall-2)

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

Figure 9. Greedy Planner Rollout to Goal (success in 10 steps)

Each panel shows step x of the greedy rollout. Colored nodes are the model’s per-hold predictions (gray = unreachable, blue = hand-only, orange = foot-only, green = both), squares mark current hand and foot contacts, and the red circle is the goal.

Figure 9 visualizes one complete greedy roll-out from the start to the goal. After each move, we recompute state-dependent features and re-run the GNN, so the colored affordance field evolves with the support set. At each step, it selects the closest holds to the end point as the next move. The figure makes clear that reachability is not a static color map. When the climber moves, the distances that feed the node features change, and the reachability field will be reconfigured. Compared with posture-aware, greedy makes faster progress with fewer detours. However, it occasionally exhibits postural imbalance and lateral displacement in sparse areas. When the geometric structure is sparser, this behavior increases the risk of failure. Last not least, it also motivates the posture-aware variant: no matter how narrow the green corridor becomes, explicit realignment steps may reduce step variability and lower failure rates on more difficult walls.

**Greedy**

图表, 条形图

AI 生成的内容可能不正确。

图表, 条形图

AI 生成的内容可能不正确。

**Posture-Aware**

图表, 条形图

AI 生成的内容可能不正确。

图表, 条形图

AI 生成的内容可能不正确。

图表, 条形图

AI 生成的内容可能不正确。

图表, 条形图

AI 生成的内容可能不正确。

图表, 条形图

AI 生成的内容可能不正确。

图表, 条形图

AI 生成的内容可能不正确。

Wall-1

Wall-2

Figure 10. Planner outcomes across walls and cohorts

Figure 10 makes the trade-offs between the two planners clear. Looking at one of the graphs separately, we can see that in terms of average number of steps completed, casual > skilled > elite (climbers in elite have better physical condition, strength, and flexibility); in terms of completion rate, it is generally elite > skilled > casual, but in the combination of wall-1 and posture-aware, the completion rate of casual exceeds skilled. When observing the same climbing wall but using different planners, the completion rates have all decreased, except for the casual completion rate of the combination of wall-1 and posture-aware, which has increased. Below are the explanations:

* The completion rate of casual increase on wall-1 with posture-aware: On wall-1, the viable path requires several small and table re-centering moves. Greedy tends to reach high and leave the feet behind, which often strands casual climbers with no legal foot candidates in the next step. The posture-aware policy checks feet below hands, angle to vertical guard, and standing reach guard to keep the body compact, realign the feet, and move the hands forward, which prevents the over-stretch dead-ends caused by greedy for short-reach. Thus, completion for casuals increases.
* The completion rate of skilled and elite decrease on wall-1 with posture-aware: Skilled and elite climbers have longer reach and can exploit transient, less-upright postures that the posture-aware policy either forbids or heavily penalizes. In our implementation, these constraints come from single-limb updates only, hands-over-feet with margin, the angle threshold that forces a foot move, and foot alignment cost. The reachable set is reduced, sometimes blocking the only feasibility sequence, so the completion rate may be lower than the greedy for skilled and elite.
* The completion rate generally decreases when switching planners: Compared to greedy, the posture-aware planner imposes stricter constraints at each step and narrows down the range of reachable fields. Any path that requires a brief deviation is more likely to be rejected, thereby increasing the failure rate. The effect is strongest on walls whose holds require lateral movement or temporary misalignment.

4.5 Ablations on Posture-Aware Policy

Tables 5-7 compare the posture-aware planner under different angle thresholds. For casual climbers, completion is best at 15-30°, then falls at 45° and drops sharply at 60°, while average steps decrease slightly. Casual climbers have shorter reach and lower strength and flexibility. A small angle realigns feet early, keeping the body upright and the hands above feet, which preserves static reach zones. With large angle, the planner postpones foot realignment, which makes the body lean out. That produces stretched and unstable postures, hence fewer steps but many more dead-ends and failures beyond 30°. For skilled climbers, completion improves as angle grows (0.71 to 0.81 to 0.90), then slips slightly at 60°. Skilled climbers can tolerate moderate tilt, which reduces foot movement, accelerates progress, and prevents immediate loss of balance. Therefore, the completion rate and efficiency improve. Pushing the angle to 60° causes some walls to exceed the stability threshold: foot movement delays become too long, hand-foot separation increases, and the frequency of standing ground contact protection mechanisms being triggered increases, which will slightly reduce the completion rate. For elite climbers, completion is high throughout and peaks at 60° and average steps decrease as angle increases. With larger arm, leg span, and flexibility, elite climbers handle big tilts safely. A large angle can make them stack multiple hand advances before realigning feet. Thanks to strength and flexibility, those stretched phases remain within static or semi-static reach and the guard rarely triggers.

Table 5. Casual: Completion Rate and Steps to Goal on Different Angle

|  |  |  |
| --- | --- | --- |
| Angle Threshold Degree | Completion Rate | Average Steps to the goal |
| 15 | 0.93 | 23.505 |
| 30 | 0.93 | 21.785 |
| 45 | 0.89 | 22.124 |
| 60 | 0.59 | 20.983 |

Table 6. Skilled: Completion Rate and Steps to Goal on Different Angle

|  |  |  |
| --- | --- | --- |
| Angle Threshold Degree | Completion Rate | Average Steps to the goal |
| 15 | 0.71 | 21.634 |
| 30 | 0.81 | 19.889 |
| 45 | 0.9 | 18.944 |
| 60 | 0.87 | 18.425 |

Table 7. Elite: Completion Rate and Steps to Goal on Different Angle

|  |  |  |
| --- | --- | --- |
| Angle Threshold Degree | Completion Rate | Average Steps to the goal |
| 15 | 0.95 | 17.589 |
| 30 | 0.97 | 16.381 |
| 45 | 0.93 | 15.538 |
| 60 | 0.99 | 15.465 |

Chapter 5

Conclusion

5.1 Project Status

5.2 Future Work

References

1. Luis-del Campo, V., Morenas Martín, J., Musculus, L. and Raab, M., 2024. Embodied planning in climbing: How pre-planning informs motor execution. *Frontiers in psychology*, *15*, p.1337878.
2. Phillips, C., Becker, L. and Bradley, E., 2012. strange beta: An assistance system for indoor rock climbing route setting. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *22*(1).
3. Nishad, S., Agarwal, S., Bhattacharya, A. and Ranu, S., 2020. Graphreach: Position-aware graph neural network using reachability estimations. *arXiv preprint arXiv:2008.09657*.
4. Stapel, F.T.A., 2020. *A heuristic approach to indoor rock climbing route generation* (Bachelor's thesis, University of Twente).
5. Vujić, S., Mirkov, D., Dikić, N., Küpper, T., Totić, S., Đoković, A., Radivojević, N., Anđelković, M., Oblaković, B.J., Baralić, I. and Manovski, K., 2021. Anthropometric, strength, endurance and flexibility characteristics of male elite ice climbers and sport climbers. *Deutsche Zeitschrift fur Sportmedizin*, *72*(2), pp.75-80.
6. Grant, S., Hynes, V., Whittaker, A. and Aitchison, T., 1996. Anthropometric, strength, endurance and flexibility characteristics of elite and recreational climbers. *Journal of sports sciences*, *14*(4), pp.301-309.
7. Laffaye, G., Levernier, G. and Collin, J.M., 2016. Determinant factors in climbing ability: Influence of strength, anthropometry, and neuromuscular fatigue. *Scandinavian journal of medicine & science in sports*, *26*(10), pp.1151-1159.
8. Draper, N., Dickson, T., Blackwell, G., Fryer, S., Priestley, S., Winter, D. and Ellis, G., 2011. Self-reported ability assessment in rock climbing. *Journal of sports sciences*, *29*(8), pp.851-858.
9. Orth, D., Davids, K. and Seifert, L., 2016. Coordination in climbing: effect of skill, practice and constraints manipulation. *Sports Medicine*, *46*(2), pp.255-268.
10. Ginszt, M., Saito, M., Zięba, E., Majcher, P. and Kikuchi, N., 2023. Body composition, anthropometric parameters, and strength-endurance characteristics of sport climbers: a systematic review. *The Journal of Strength & Conditioning Research*, *37*(6), pp.1339-1348.
11. Giles, D., Barnes, K., Taylor, N., Chidley, C., Chidley, J., Mitchell, J., Torr, O., Gibson-Smith, E. and España-Romero, V., 2021. Anthropometry and performance characteristics of recreational advanced to elite female rock climbers. *Journal of sports sciences*, *39*(1), pp.48-56.
12. Seifert, L., Wattebled, L., Herault, R., Poizat, G., Adé, D., Gal-Petitfaux, N. and Davids, K., 2014. Neurobiological degeneracy and affordance perception support functional intra-individual variability of inter-limb coordination during ice climbing. *PloS one*, *9*(2), p.e89865.
13. España-Romero, V., Ortega Porcel, F.B., Artero, E.G., Jiménez-Pavón, D., Gutiérrez Sainz, Á., Castillo Garzón, M.J. and Ruiz, J.R., 2009. Climbing time to exhaustion is a determinant of climbing performance in high-level sport climbers. *European journal of applied physiology*, *107*(5), pp.517-525.
14. Fuss, F.K. and Niegl, G., 2010. Biomechanics of the two-handed dyno technique for sport climbing. *Sports Engineering*, *13*(1), pp.19-30.
15. Morrison, A.B. and Schöffl, V.R., 2007. Physiological responses to rock climbing in young climbers. *British journal of sports medicine*, *41*(12), pp.852-861.
16. White, D.J. and Olsen, P.D., 2010. A time motion analysis of bouldering style competitive rock climbing. *The Journal of Strength & Conditioning Research*, *24*(5), pp.1356-1360.
17. Hamilton, W., Ying, Z. and Leskovec, J., 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, *30*.
18. Ardón, P., Pairet, È., Lohan, K.S., Ramamoorthy, S. and Petrick, R., 2020. Affordances in robotic tasks--a survey. *arXiv preprint arXiv:2004.07400*.
19. Kipf, T.N., 2016. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*.
20. Perez, L. and Wang, J., 2017. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*.
21. Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A. and Vandergheynst, P., 2017. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, *34*(4), pp.18-42.
22. Quaine, F., Martin, L. and Blanchi, J.P., 1997. The effect of body position and number of supports on wall reaction forces in rock climbing. *Journal of Applied Biomechanics*, *13*(1), pp.14-23.
23. Seifert, L., Orth, D., Button, C., Brymer, E. and Davids, K., 2016. An ecological dynamics framework for the acquisition of perceptual–motor skills in climbing. In *Extreme sports medicine* (pp. 365-382). Cham: Springer International Publishing.
24. Mermier, C.M., Janot, J.M., Parker, D.L. and Swan, J.G., 2000. Physiological and anthropometric determinants of sport climbing performance. *British journal of sports medicine*, *34*(5), pp.359-365.
25. Seifert, L., Cordier, R., Orth, D., Courtine, Y. and Croft, J.L., 2017. Role of route previewing strategies on climbing fluency and exploratory movements. *PloS one*, *12*(4), p.e0176306.
26. Hitka, M., Gejdoš, M. and Sydor, M., 2024. Dataset on anthropometric measurements of the adult population in Slovakia. *Data in Brief*, *55*, p.110536.
27. Baláš, J., Pecha, O., Martin, A.J. and Cochrane, D., 2012. Hand–arm strength and endurance as predictors of climbing performance. *European journal of sport science*, *12*(1), pp.16-25.
28. Gordon, J., Maselli, A., Lancia, G.L., Thiery, T., Cisek, P. and Pezzulo, G., 2021. The road towards understanding embodied decisions. *Neuroscience & Biobehavioral Reviews*, *131*, pp.722-736.
29. Pascanu, R., Mikolov, T. and Bengio, Y., 2013, May. On the difficulty of training recurrent neural networks. In *International conference on machine learning* (pp. 1310-1318). Pmlr.
30. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. and Bengio, Y., 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.
31. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O. and Dahl, G.E., 2017, July. Neural message passing for quantum chemistry. In *International conference on machine learning* (pp. 1263-1272). Pmlr.
32. Battaglia, P.W., Hamrick, J.B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R. and Gulcehre, C., 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.
33. Bronstein, M.M., Bruna, J., Cohen, T. and Veličković, P., 2021. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*.
34. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).
35. Xu, K., Hu, W., Leskovec, J. and Jegelka, S., 2018. How powerful are graph neural networks?. *arXiv preprint arXiv:1810.00826*.
36. Lin, T.Y., Goyal, P., Girshick, R., He, K. and Dollár, P., 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980-2988).
37. Cui, Y., Jia, M., Lin, T.Y., Song, Y. and Belongie, S., 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9268-9277).
38. Smith, L.N., 2017, March. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)* (pp. 464-472). IEEE.
39. Fey, M. and Lenssen, J.E., 2019. Fast graph representation learning with PyTorch Geometric. *arXiv preprint arXiv:1903.02428*.