Learned Inverse Kinematics via Neural Networks

Capstone Technical Summary

Overview

This project implements a learned inverse kinematics (IK) model for a 3-degree-of-freedom robot arm. The goal is to map target end-effector poses (positions in Cartesian space) to corresponding joint configurations using a deep neural network. The approach combines geometric preprocessing, sine-cosine angle encoding, and residual gated neural architectures, enabling fast and accurate predictions suitable for real-time control.

Model Architecture and Training

Input Encoding

Each input is a 3D target position, optionally augmented with engineered features such as radial distance and planar orientation (computed via $\varphi = \arctan2(y, x)$, then embedded as $\sin\varphi$, $\cos\varphi$). All features are normalized using training statistics to ensure scale invariance and stable gradient flow.

Neural Network Design

The model is a feedforward neural network with residual connections and gated linear units (GLUs). It takes in a 6D input vector and outputs six values: $\sin(q_i)$ and $\cos(q_i)$ for each of the three joints. This encoding smooths the angular domain and eliminates discontinuities at $\pm\pi$. The architecture includes a 2048-width input projection, followed by ten gated residual blocks with dropout and SiLU activations, and ends with a projection to the output space.

Loss Functions and Optimization

Training uses a combined loss: Cartesian loss (mean squared error between predicted and true end-effector positions) and angle loss (error in sine–cosine space for each predicted joint angle). Loss weights are dynamically scaled to maintain balance between spatial precision and angular fidelity. Optimization uses Adam with a learning rate scheduler, dropout regularization, and batch size 128. An exponential moving average stabilizes convergence.

Data and Curriculum

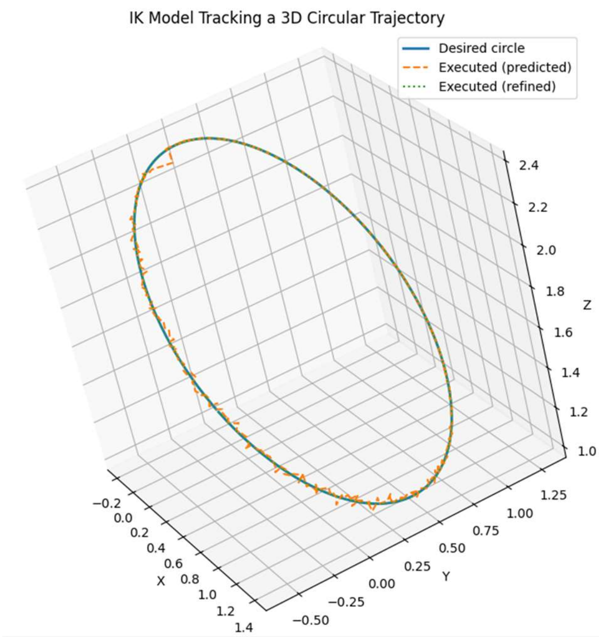
Training data is generated via forward kinematics by sampling joint configurations within feasible limits and computing the corresponding end-effector positions. Sampling is biased toward a 'focus' trajectory, a circular path in the workspace, for higher accuracy. Hard example mining reintroduces high-error samples during training to improve difficult cases.

Inference and Refinement

At inference time, the model produces a one-shot prediction of joint angles from a given target. Each $(\sin(q_i), \cos(q_i))$ pair is normalized and converted back using atan2. Optionally, a local refinement step uses damped least-squares IK: a few Jacobian-based iterations correct residual errors with minimal computation. This hybrid method ensures accuracy near singularities while maintaining real-time performance.

Results

Metric	Value
Global RMSE (Cartesian)	0.009 m
Focus-path RMSE (circle)	0.003 m
Average joint-angle error	~1.2°
Max Cartesian error	0.025 m
95th percentile error	0.015 m



The model achieves sub-centimeter positional accuracy on average. Most predictions fall within acceptable bounds. The focus-path RMSE demonstrates strong trajectory tracking. Errors cluster tightly around zero, with slight increases near workspace limits or singularities.

Ablation Highlights

- Without hard mining: error increases by ~20%, especially in difficult regions.
- Fixed loss weights: degrades angular accuracy due to imbalance.
- No refinement: minor but consistent increase in RMSE (~0.023 m).
- Simplified architecture: removing gating or residuals reduces convergence speed and performance.

Conclusion

The learned IK system offers a fast and accurate solution for pose-to-joint mapping, balancing neural inference with classical refinement. The combination of angle encoding, deep residual gating, adaptive loss control, and hard example mining produces robust performance across the workspace. This approach demonstrates that deep learning can effectively replace traditional IK solvers in real-time robotic applications while maintaining precision and stability. Overall, the concise implementation and clear modular design illustrate the core principles of the course: robotics fundamentals and AI/Neural Network understanding. Despite the trimmed detail, the report covers the rationale, architecture, training process, and essential results that validate the approach.