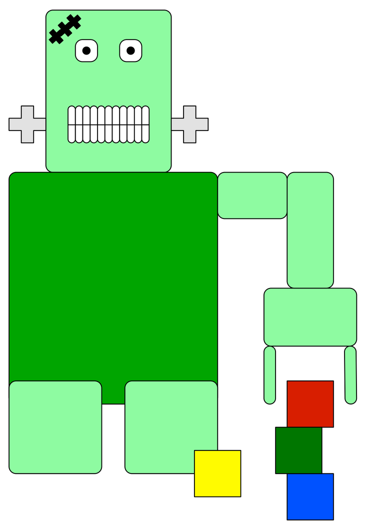


Training Frankenstein's Creature to Stack: HyperTree Architecture Search

Automate the design of deep neural network architectures for robotics.

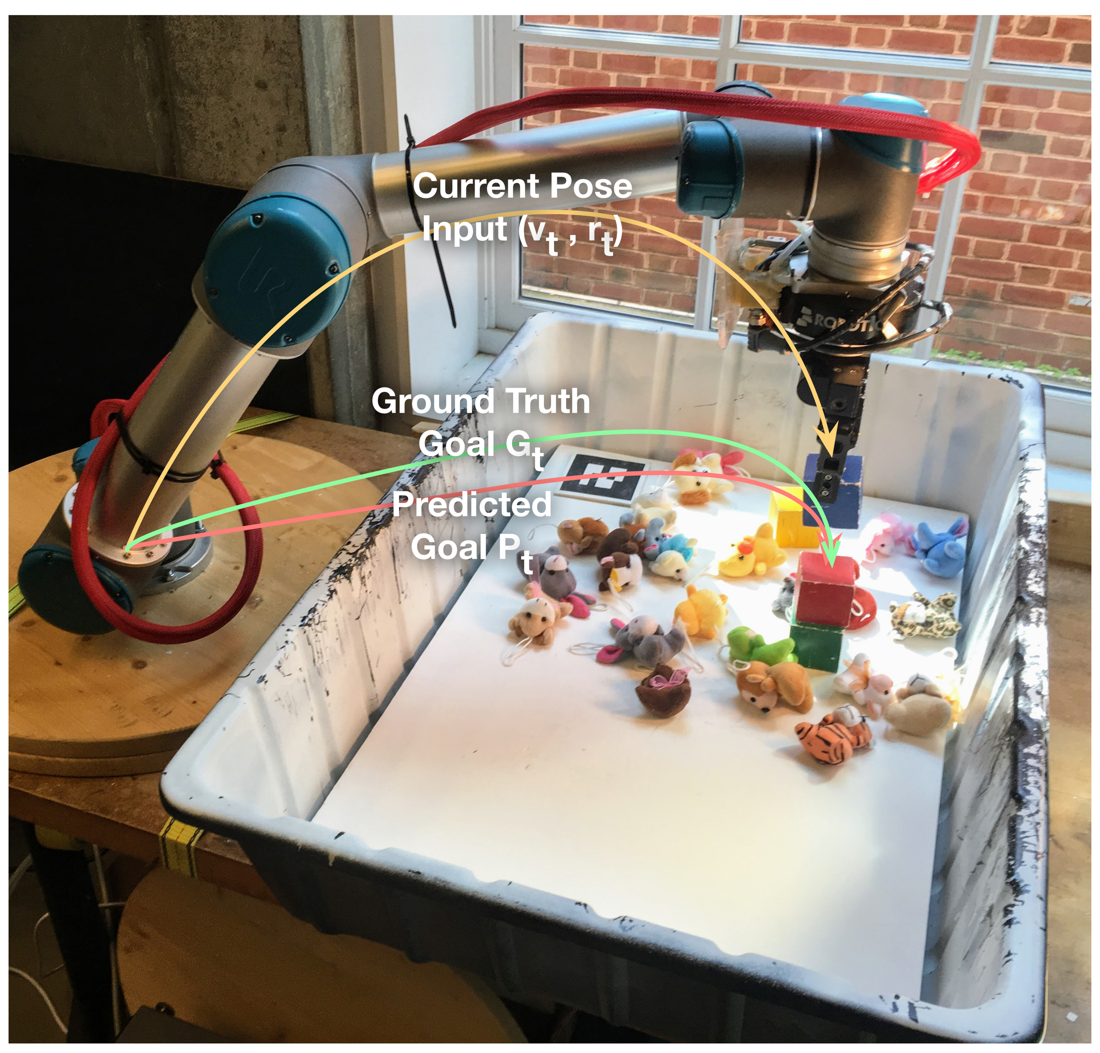
JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

Andrew Hundt <ahundt@jhu.edu>, Varun Jain, Chris Paxton, Gregory D. Hager

This material is based upon work supported by the National Science Foundation under NRI Grant Award No. 1637949.

CoSTAR Block Stacking Dataset

The CoSTAR Block Stacking Dataset includes a real robot trying to stack colored children's blocks more than 10,000 times. It is designed to benchmark neural network based algorithms.



Includes:

- vastly different lighting conditions
- plush toy distractors
- stacks of 3 or 4 blocks
- object wear
- movable bin obstacle which must be avoided
- successes and failures

Current Version: v0.4		Last updated: 09/25/2018
Overview		
Calibrated Images	Color, depth (resolution 640x480)	
Joint Data	Angle (radian), velocity (radians/s)	
Typical Duration	20 seconds, 200 frames, 100ms per frame (10 Hz)	
Labels	stack success/failure/error, action name	
3D Coordinate Poses Recorded		
Gripper	RGB camera	Depth camera
Robot joints	AR tags and ID#	Colored blocks
Statistics		
	Blocks	**Blocks and Toys**
Attempts	5884	6106
Success	2451	748
Failures, all kinds	3433	5358
Failures without errors	1233	3628
Failures with errors	2200	1703
Success only subset		
Training	2195	620
Validation	128	64
Test	128	64

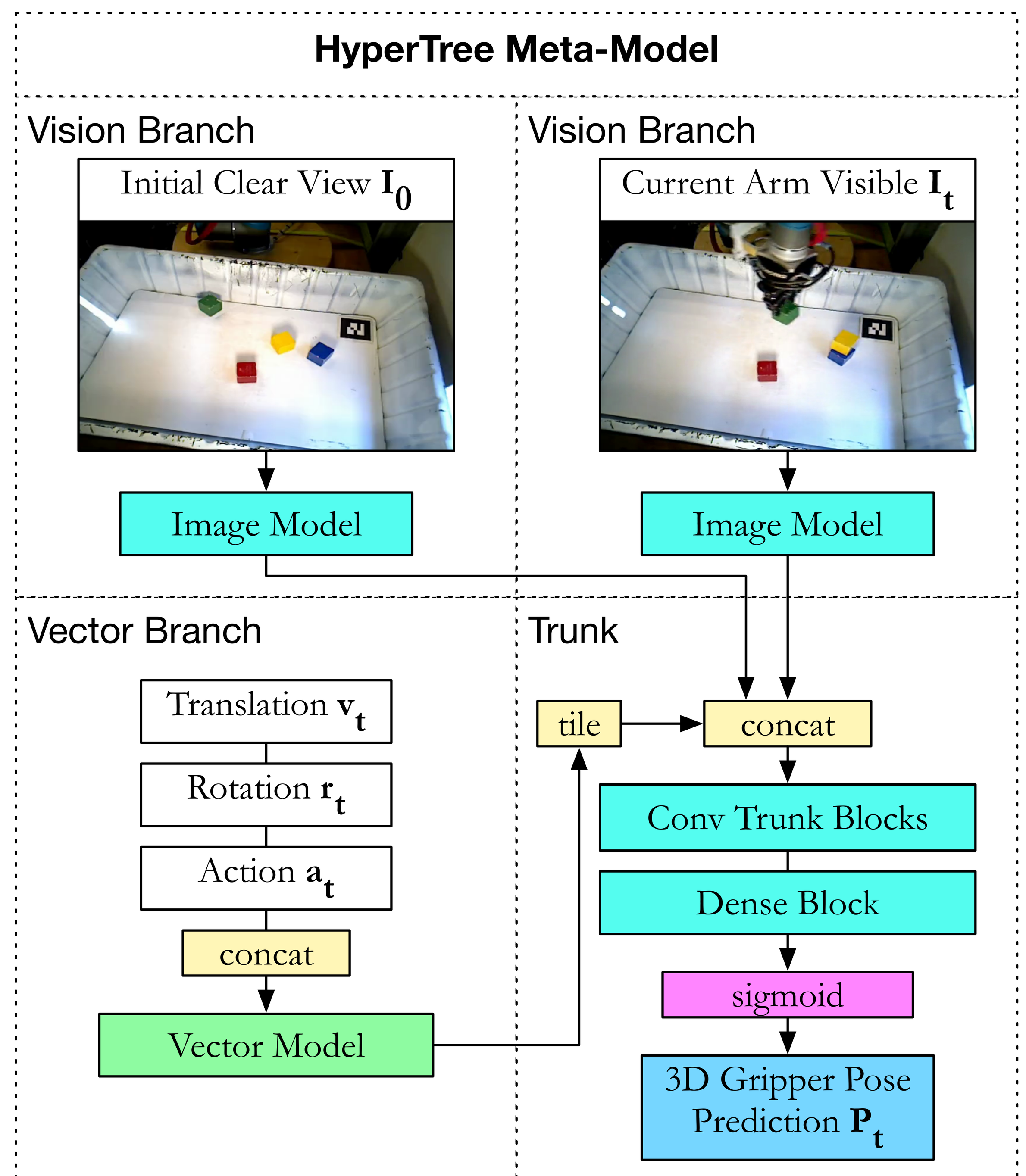
Each row shows key goal time steps from separate stacking attempts. Images sequences are ordered from left to right.

Dataset Videos and Details:
sites.google.com/site/costardataset

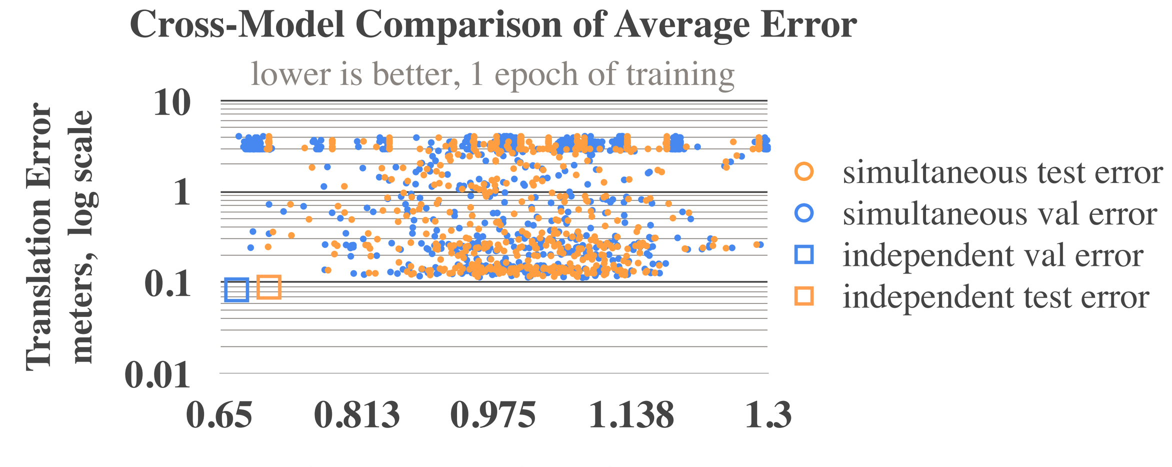
Abstract, Videos, and Paper:
sites.google.com/site/hypertree-renas

HyperTree Architecture Search

Low cost automatic design of multiple-input neural network models with Baysesian Optimization.



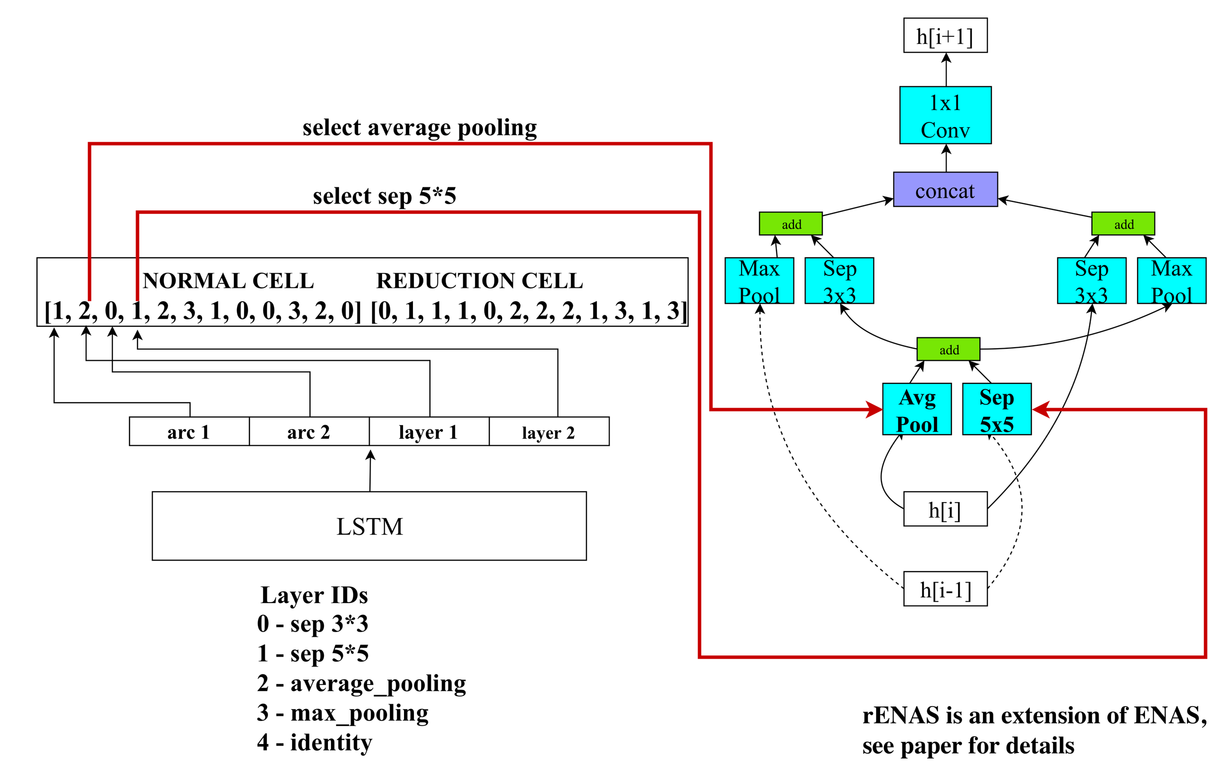
Much like how Dr. Frankenstein's creature was assembled from pieces before he came to life in the eponymous book, HyperTrees substitute in and combine parts of other architectures to optimize for a new problem domain. Particular component substitution details can be found in the paper.



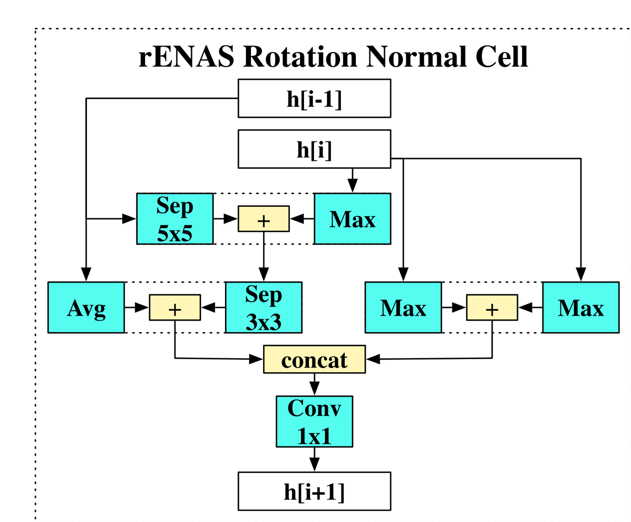
Predicting translation and rotation of the gripper independently was more accurate than making those predictions simultaneously. Each mark is a separate HyperTree model with 1 epoch of training.

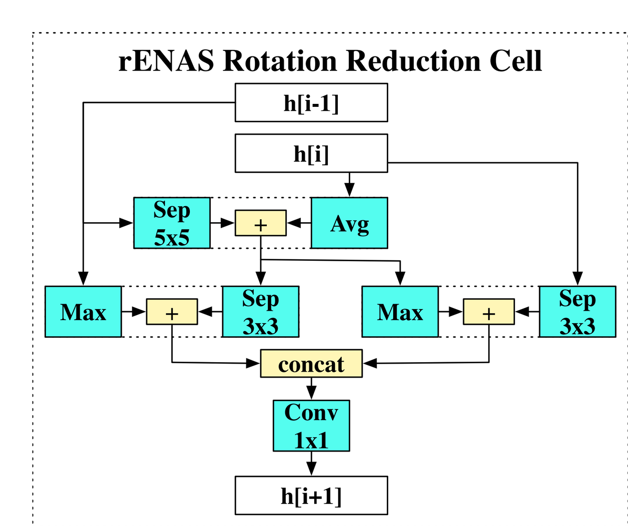
rENAS: regression Efficient Neural Architecture Search

Low cost automatic design of multiple-input neural network models with Reinforcement Learning.



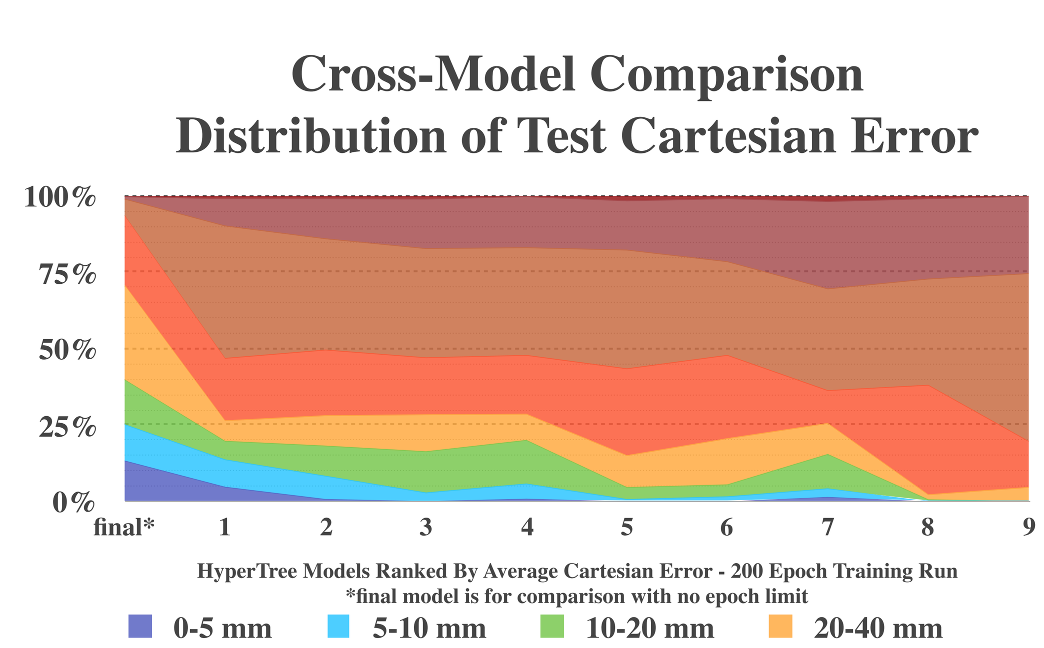
- An LSTM predicts architectures in a meta-model
- Weights are not discarded, increasing search efficiency
- rENAS extends the so-called "micro search space" of ENAS with a new loss and reward function to minimize error.
- rENAS parameterizes placement and number of reduction cells, which rescale the data width and height by half.

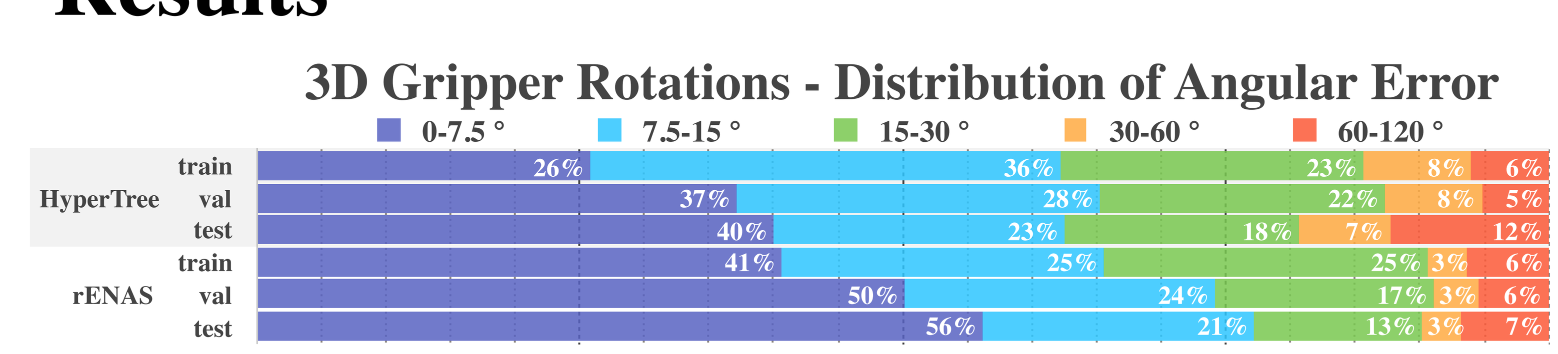


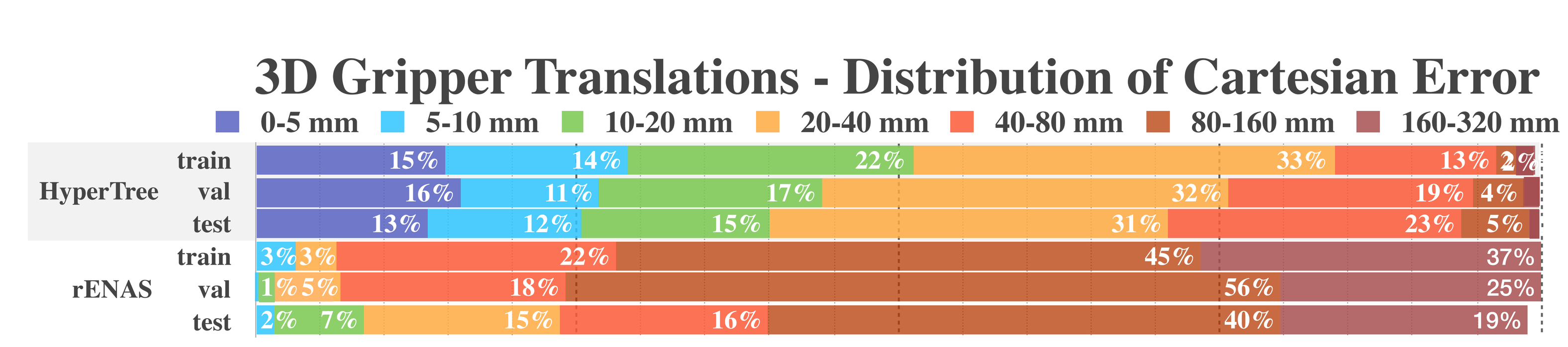


Final rENAS Rotation Cells

Results







A high percentage of samples with low error is better. Results compare the predicted gripper positions and orientations against the real robot data in the CoSTAR Block Stacking Dataset. This is done by showing the neural network random time steps in the video and asking it to predict the position and orientation the robot will have at the next goal. **(Left)** The importance of hyperparameter choice is visible in models 1-9 which were selected from the best of 1100 HyperTree candidates and then trained for 200 epochs.