2020 embedded VISION summit®

Imaging Systems for Applied Reinforcement Learning Control

Damas Limoge – Sr. R&D Engineer
Nanotronics
September 2020

nanotronics

To build the future, you need to see it.

-Matthew Putman, CEO

nanotronics

Outline

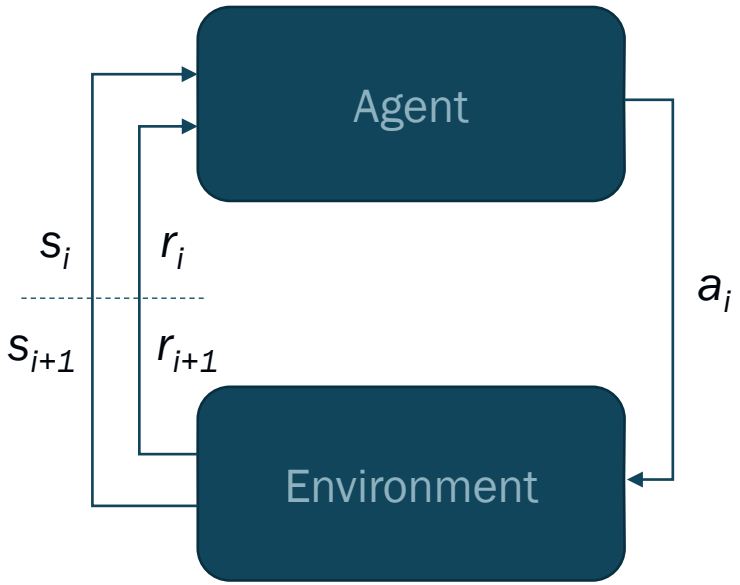
1. Introduction to Reinforcement Learning
2. Application Areas for Reinforcement Learning
3. Reinforcement Learning for Imaging Systems
4. Challenges for Using Reinforcement Learning
5. Case Study – Additive Manufacturing Feedback using Image Data
6. Insights and Conclusions

Introduction to Reinforcement Learning

nanotronics

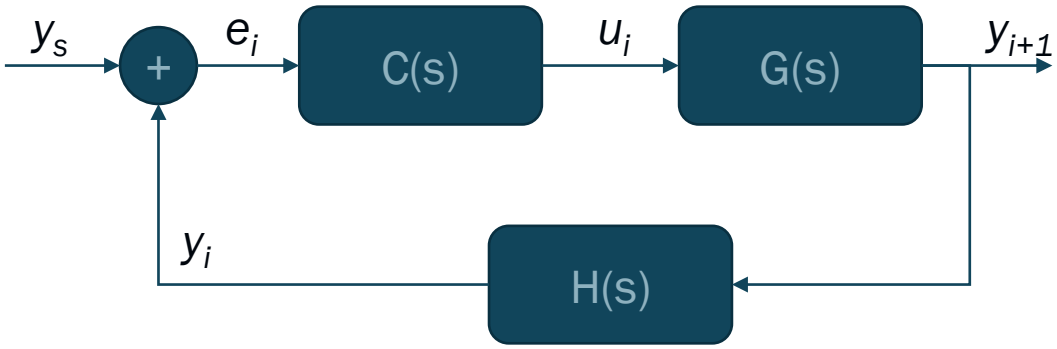
Control Feedback Loops

Reinforcement Learning Loop



(state, action, reward)

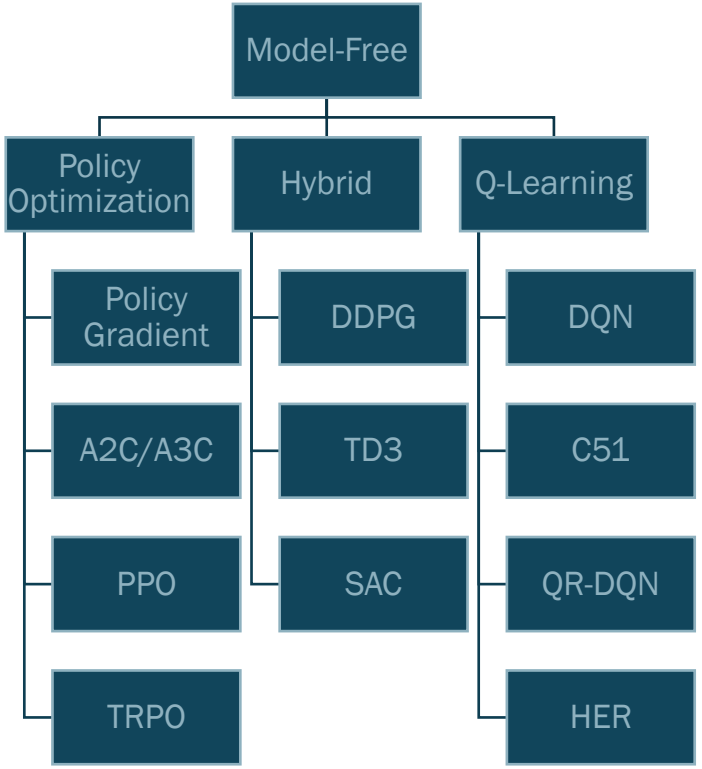
Classical Control Loop

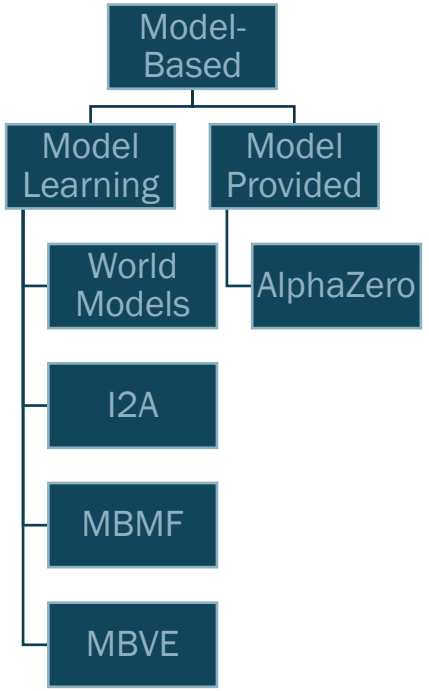


(error, effort, output)

Reinforcement Learning Algorithms

Given the volume of options, choosing the appropriate algorithm for the problem statement is key.

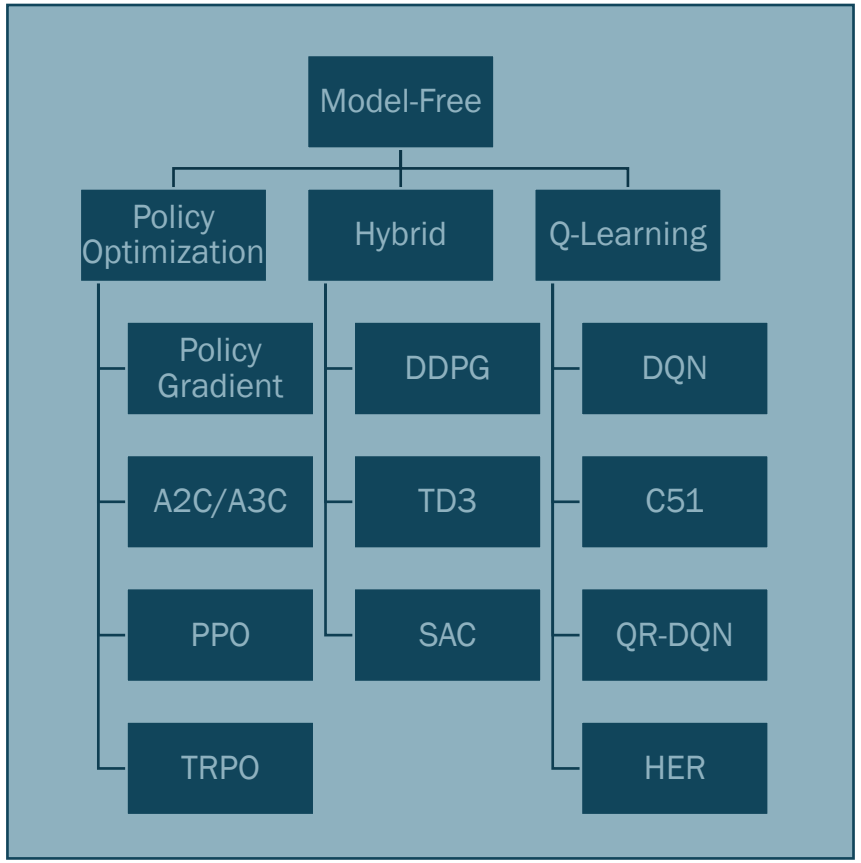


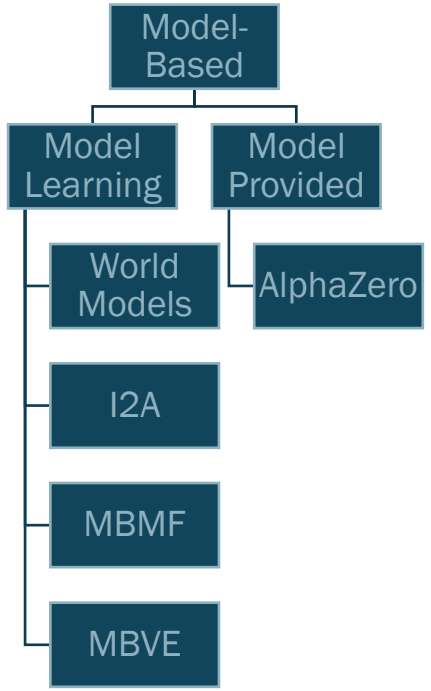


Source: https://spinningup.openai.com/en/latest/spinningup/rl_intro2.html

Reinforcement Learning Algorithms

We'll focus on the model-free variety, as imaging systems exist in noisy, difficult-to-model contexts.





Source: https://spinningup.openai.com/en/latest/spinningup/rl_intro2.html

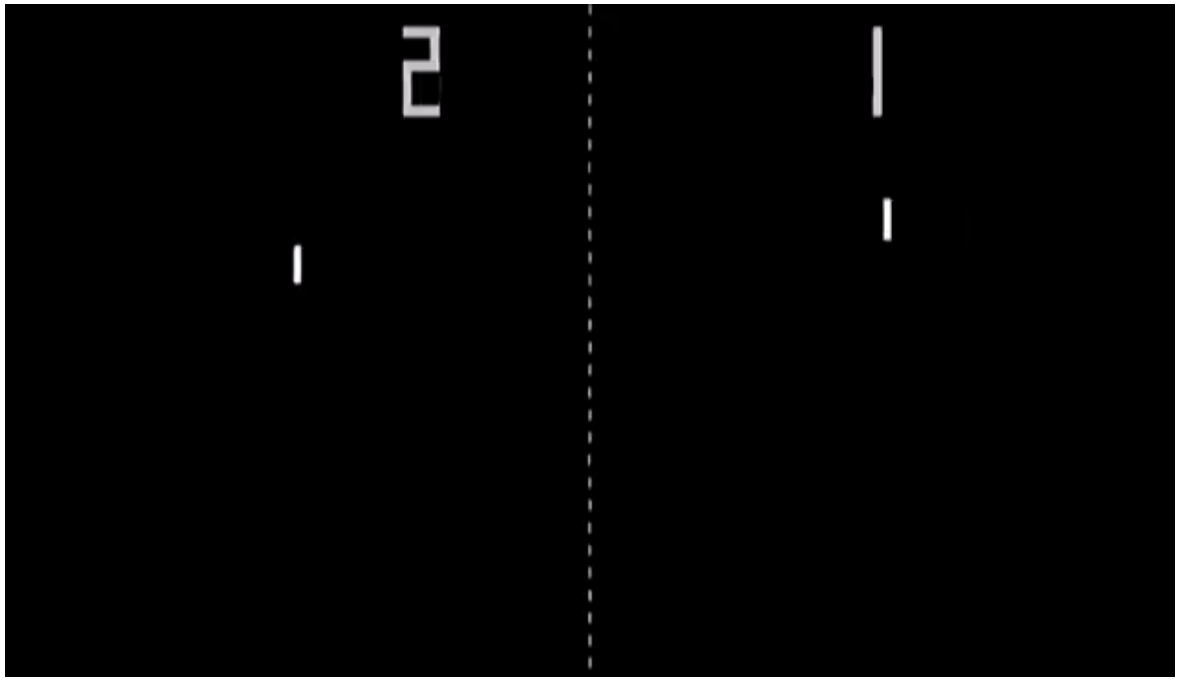
Explanation of Model-Free Reinforcement Learning

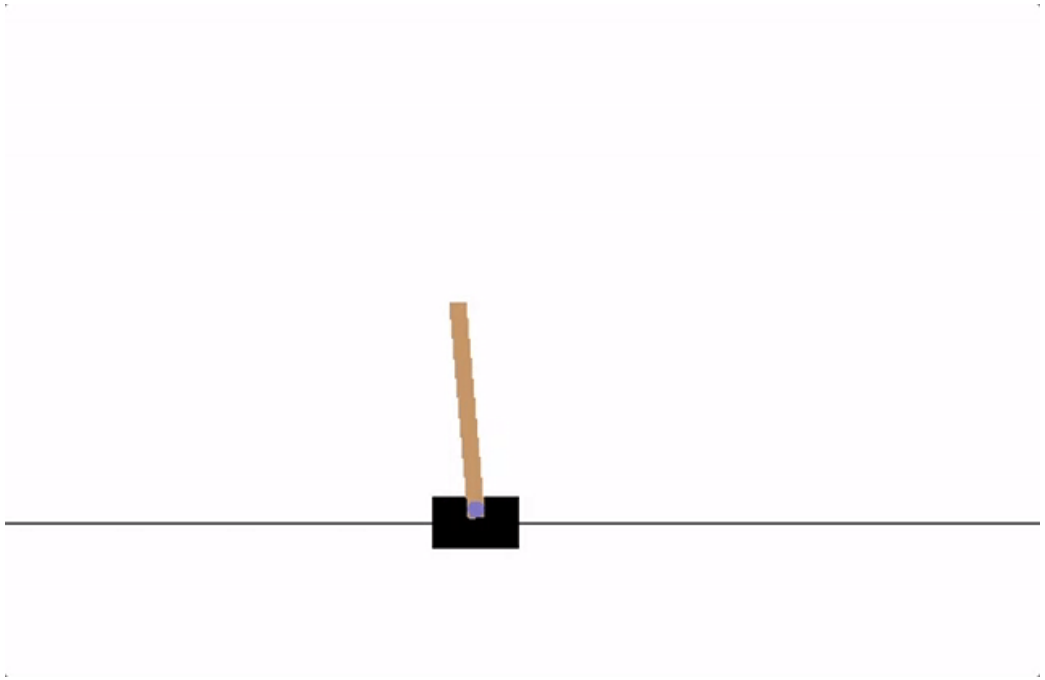
- Model-Free reinforcement learning allows us to learn an environment as we traverse it.
- If we have an imperfect representation of the model to act on, that is okay, as we anticipate improving our strategy as we continue.
- This plays well with physical measurements as it requires less exact measurements for a prediction of optimal actions.
- Reinforcement learning in general *does not* play well with real, physical environments because it requires a huge volume of training examples.
    - Reducing the required training examples is a huge focus of research in reinforcement learning, and a particular focus in physical applications.
    - Imaging systems are a subset of physical environments. An imaging system is meant to capture rich, albeit abstract representations of that environment. The RL algorithm is meant to map that to a policy for choosing optimal actions within the environment.

Application Areas for Reinforcement Learning

nanotronics

Simple Environments

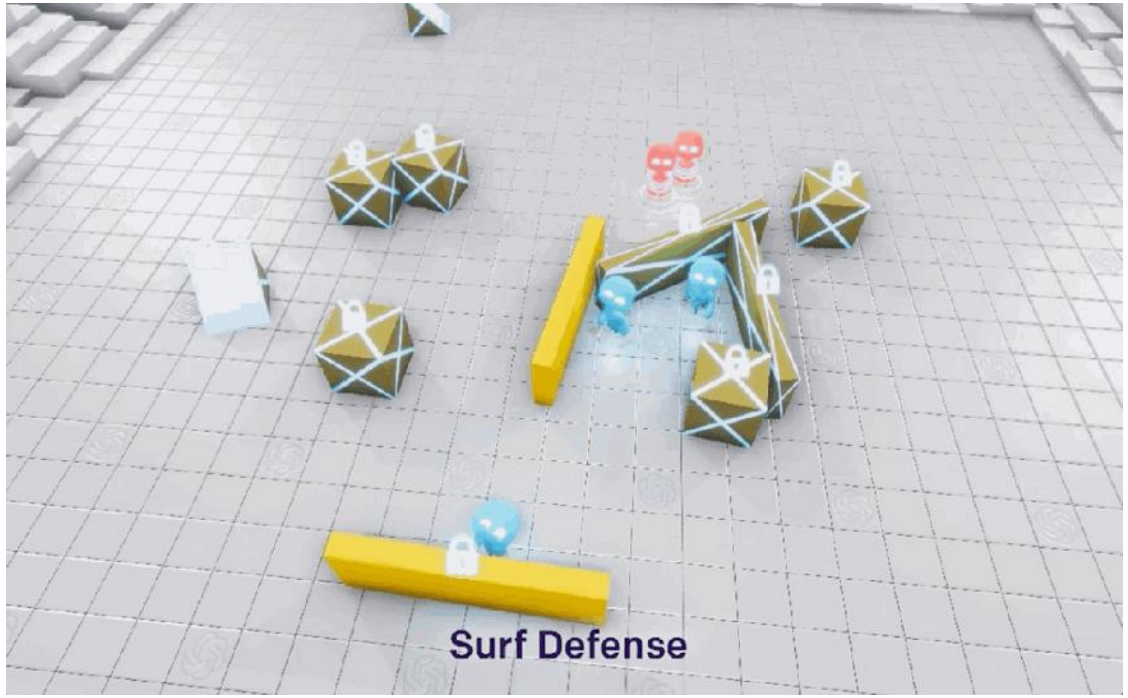




Sources:

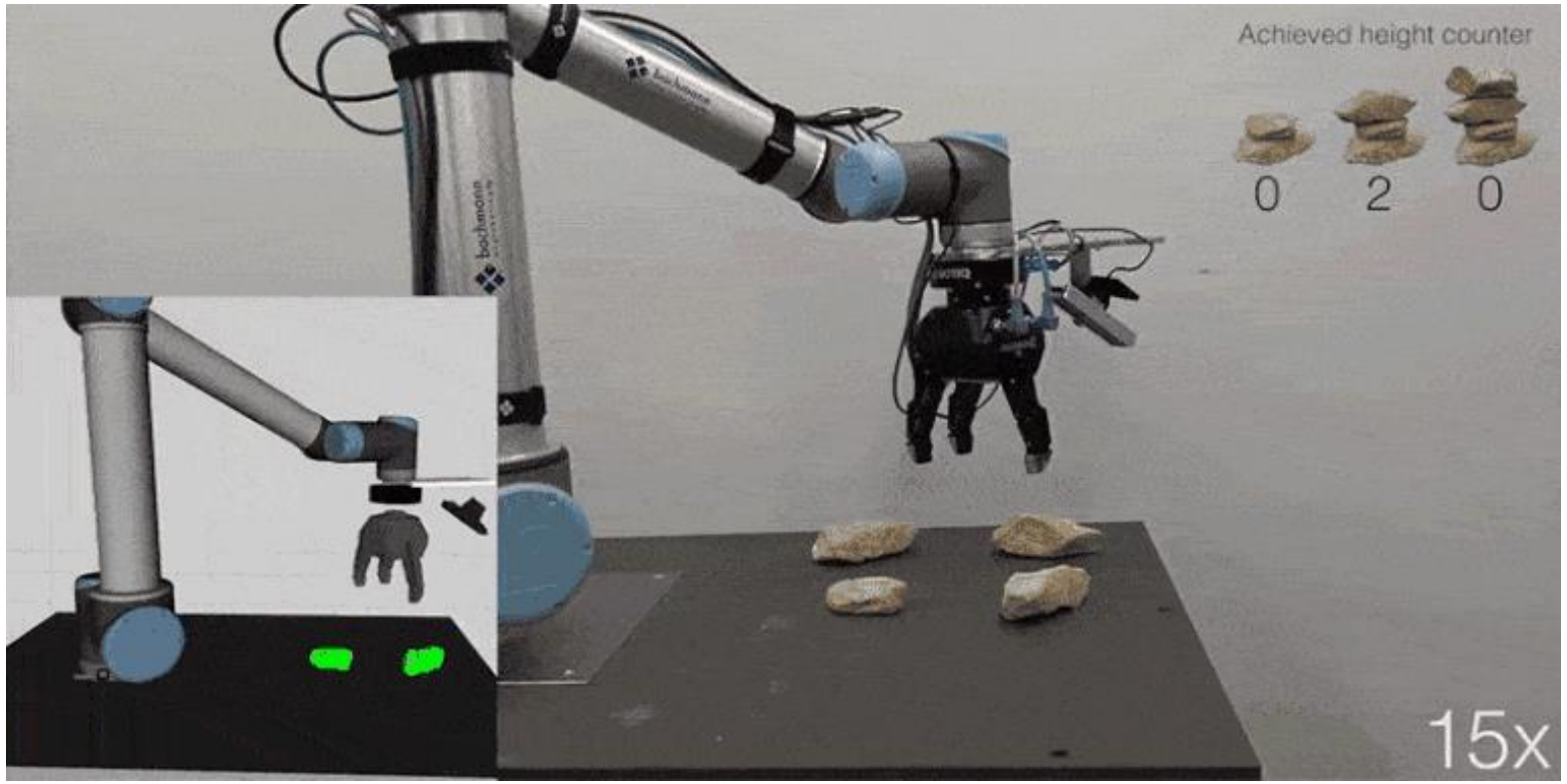
1. (Left) Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." arXiv preprint arXiv:1312.5602 (2013).
2. (Right) https://gym.openai.com/envs/CartPole-v0/

Complex Environments



Sources:
1. (Left) Berner, Christopher, et al. "Dota 2 with large scale deep reinforcement learning." arXiv preprint arXiv:1912.06680 (2019).
2. (Right) Baker, Bowen, et al. "Emergent tool use from multi-agent autocurricula." arXiv preprint arXiv:1909.07528 (2019).

Vision Based Systems



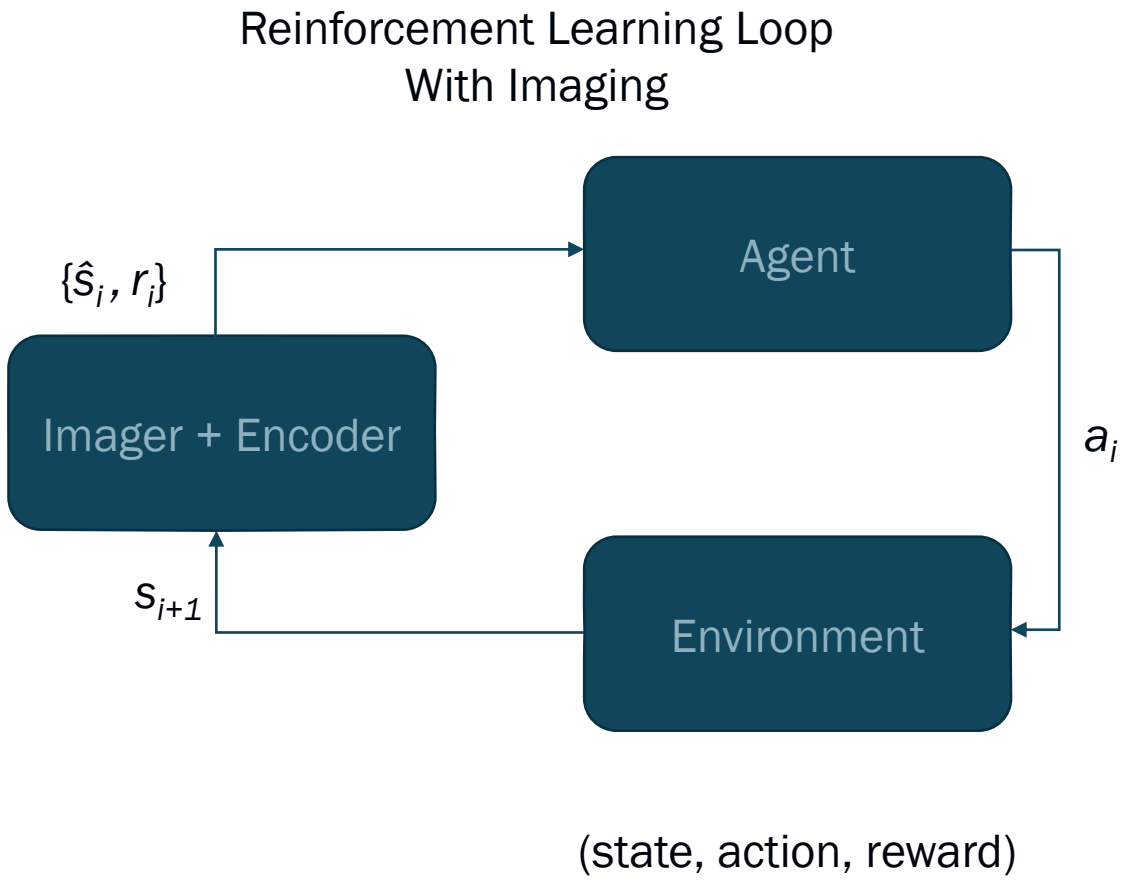
Source:

1. https://gizmodo.com/rock-balancing-robots-could-build-our-future-habitats-o-1795814087

Reinforcement Learning for Imaging Systems

nanotronics

Adjustment to the Reinforcement Learning Structure



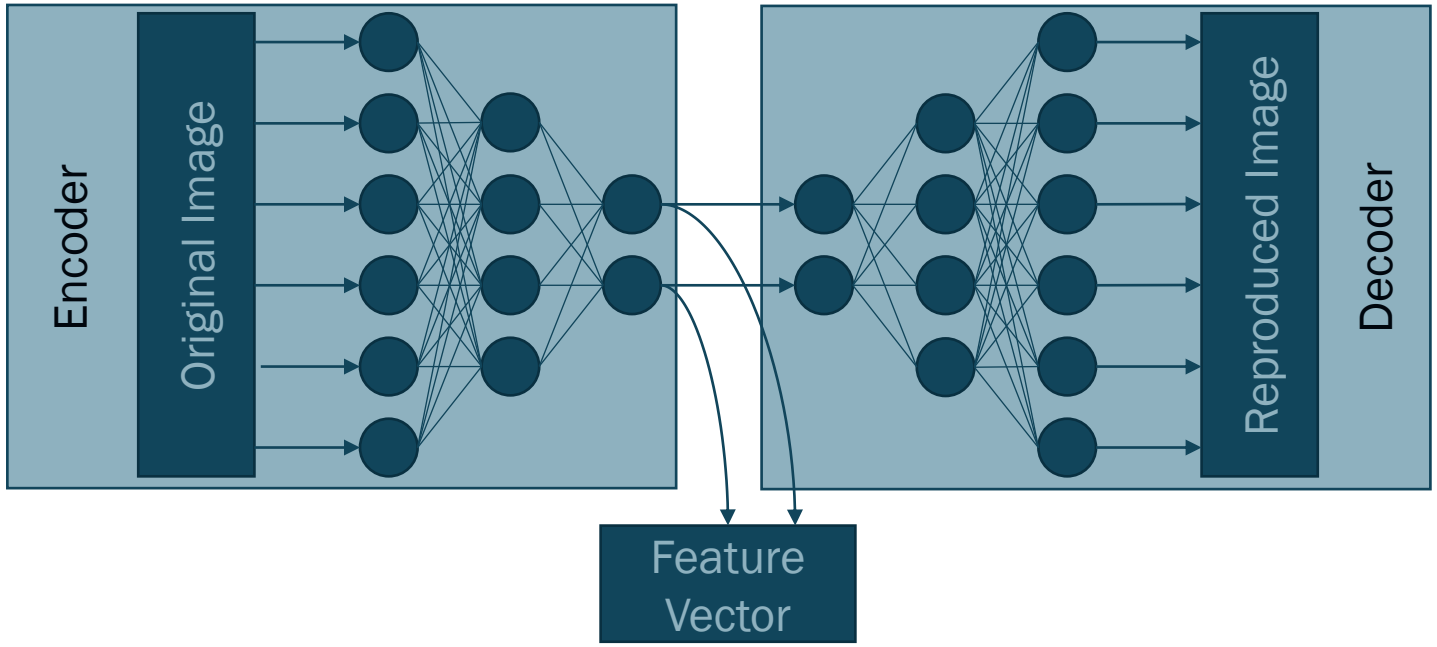
- The environment must be effectively sampled by an image acquisition system.
- This state must include the requisite information for control observability.
- The dimension of this state must be reduced from the densely complex image state to a reduced dimensional representative encoding.

State Embedding

- For most image-based machine learning, pixel values of images are fed into convolutional layers. The output of these layers are called the features of the image.
- Feature vectors are fundamentally the states of image-based reinforcement learning as well, but the training of their convolutional weights can be prohibitively costly without GPU clusters.
- This is distinct from other image classifiers in that a very clear path exists for determining, from a static set of examples, the appropriate network architecture and adjust its dimensions using A/B testing.
- In RL applications, it may be prudent to try several different architectures, each requiring subsequent tuning. For this reason, reducing the dimensionality of the input state is helpful in faster iteration through model types.

State Embedding (cont'd)

- We can leverage unsupervised methods to train an encoder out of the loop and use scalar targets or human-led instruction for ensuring essential information is maintained.
- This capitalizes on the speed with which conventional classifiers can be trained, while allowing flexibility of the network architecture for the agent.
- It is analogous to compressing the image and using only a combination of values which provide the highest information content.
- Furthermore, we can target certain areas or features by building a loss function that penalizes low information through-put for our desired content.



Challenges for Using Reinforcement Learning

nanotronics

Challenges in Physical Environments

- Reinforcement learning in physical environments suffers all of the same problems, but training samples are billions of times more temporally expensive to generate.
- Physical data is often noisy and actuation setpoints are often imprecise.
- The underlying dynamics of physical systems have been well studied in the field of controls, and thus the application of reinforcement learning is often meant to replace a system that can't be readily modeled or controlled.
- Physical environments suffer more severe consequences for unfettered agent exploration, involving damage to equipment or observers.

Reduction of Combinational Complexity

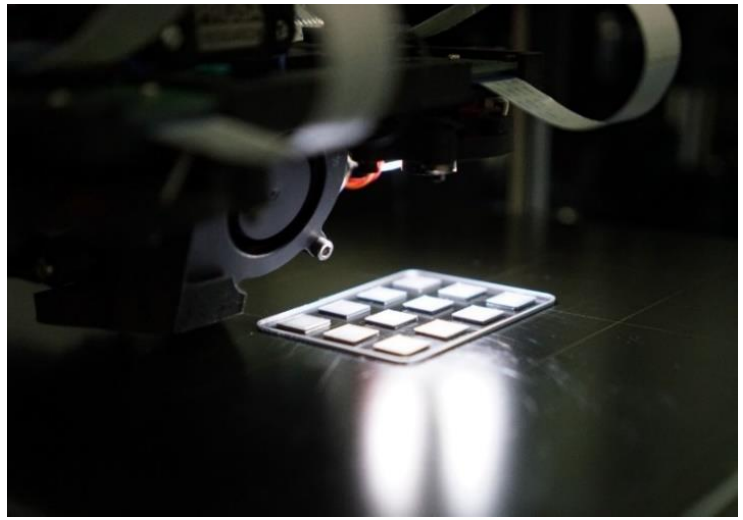
- The initial challenge to overcome in physical systems is the acquisition of sufficient data samples to train the agent. One billion samples, even for reasonably fast processes, often extends past the span of many lifetimes.
- However, some reinforcement algorithms applied to simple environments with appropriately scaled networks have shown convergence on the order of hundreds or thousands of samples. While these are still costly volumes in some contexts, they are not untenable at face value.
- The key to achieving this type of sample efficiency is appropriately defined problems to only consider the actuation points that solve the reward, and likewise a minimized state definition for consideration by the agent.

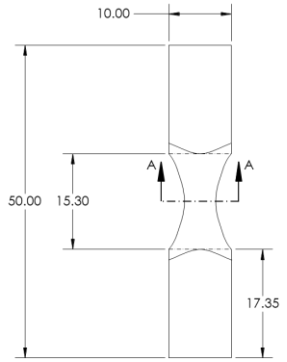
Case Study:
Additive Manufacturing Feedback using Image Data

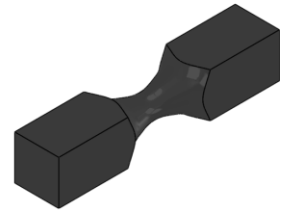
nanotronics

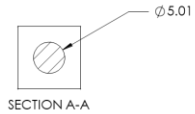
Case Study Overview

- Additive Manufacturing is a burgeoning field of manufacturing that uses material deposition, curing or fusion to build geometries that are unattainable with other fabrication methods. It is commonly referred to as 3D Printing.
- Its major hurdle for transitioning from prototype use-cases to industrial applications is, among other things, inconsistency of the material properties after printing. The inconsistency is caused by printing errors such as improper material deposition
- The aim of this case study was to build a part (shown to the right) with the most *consistent* tensile strength. The part was designed to be susceptible to printing errors in the cross-sectional plane.
- The allowable actions corresponded to adjusting the material flow and the speed of the print.
- Two test cases were considered for RL feedback using Images:
    - Discrete Error Correction
        - An error was injected at a specific layer, and the agent was allowed to make corrections on three subsequent layers.
        - The layer number as well as the error as classified by an image classifier was used as the state definition.
    - Continuous Error Correction
        - No errors were injected into the prints, but instead corrections were allowed at all layers.
        - The state was defined as the output of an autoencoder trained on a two-dimensional Gaussian loss function centered around the layer image.

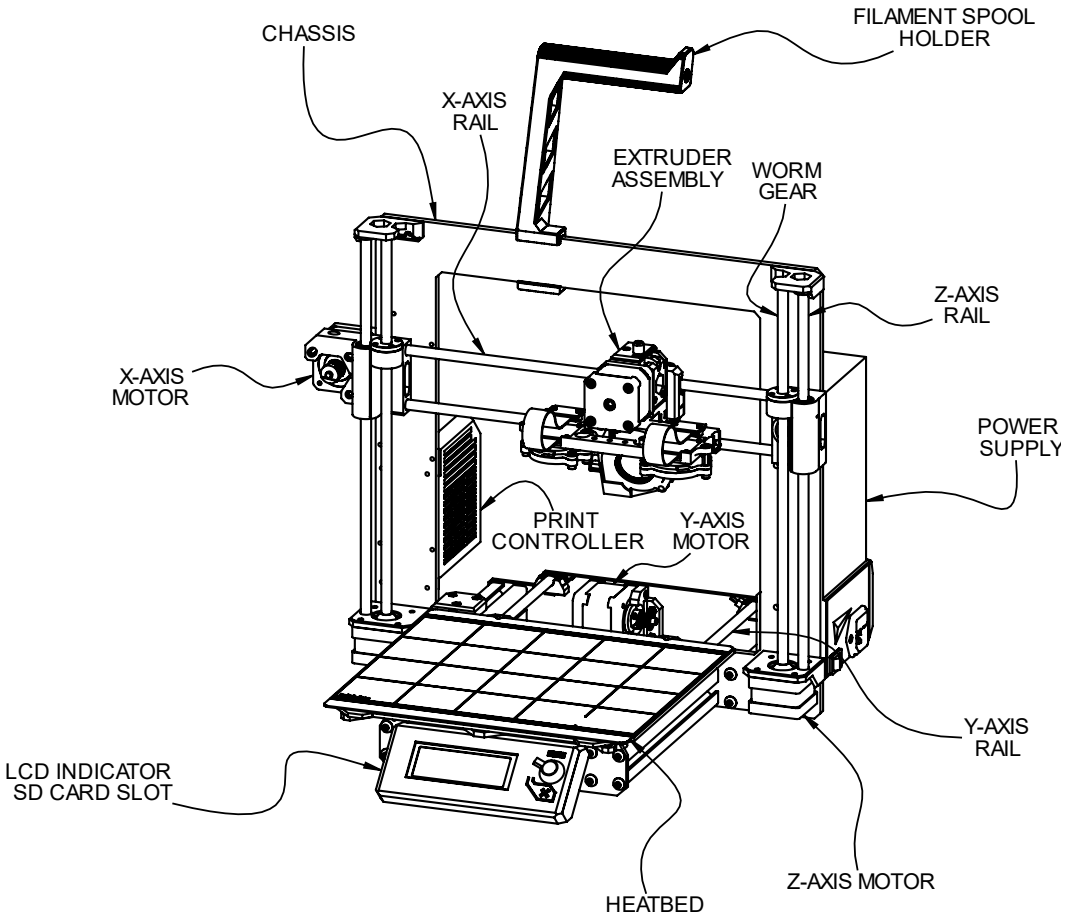


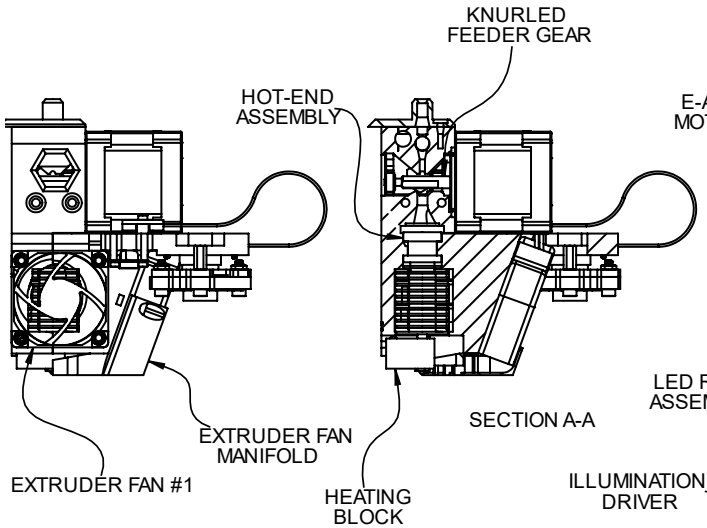


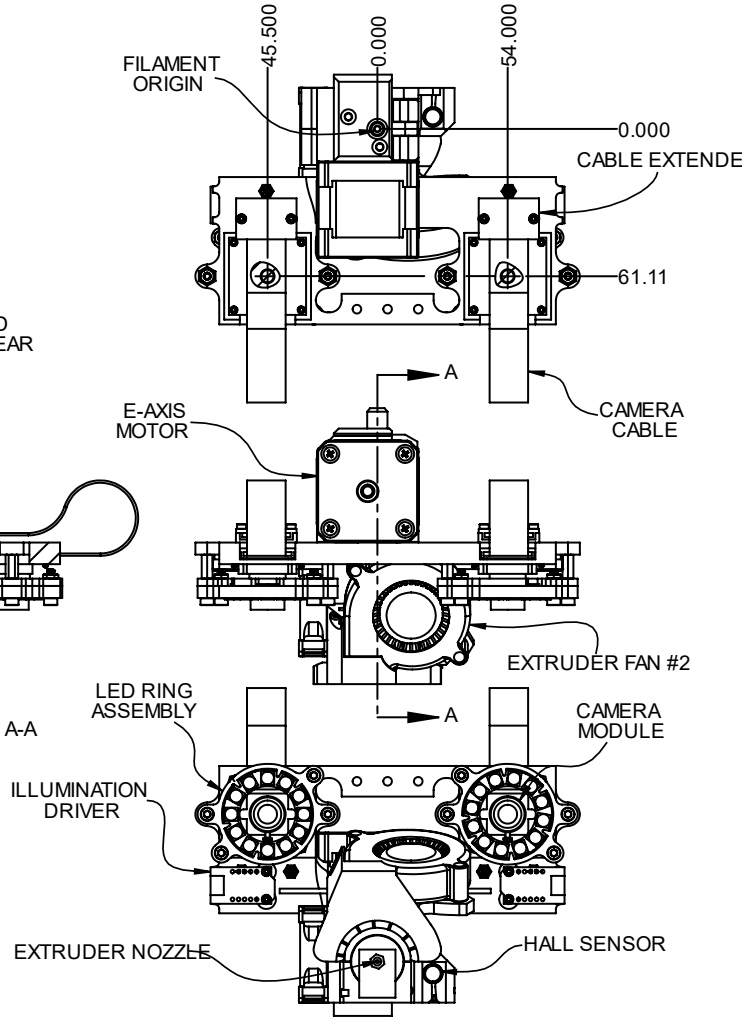




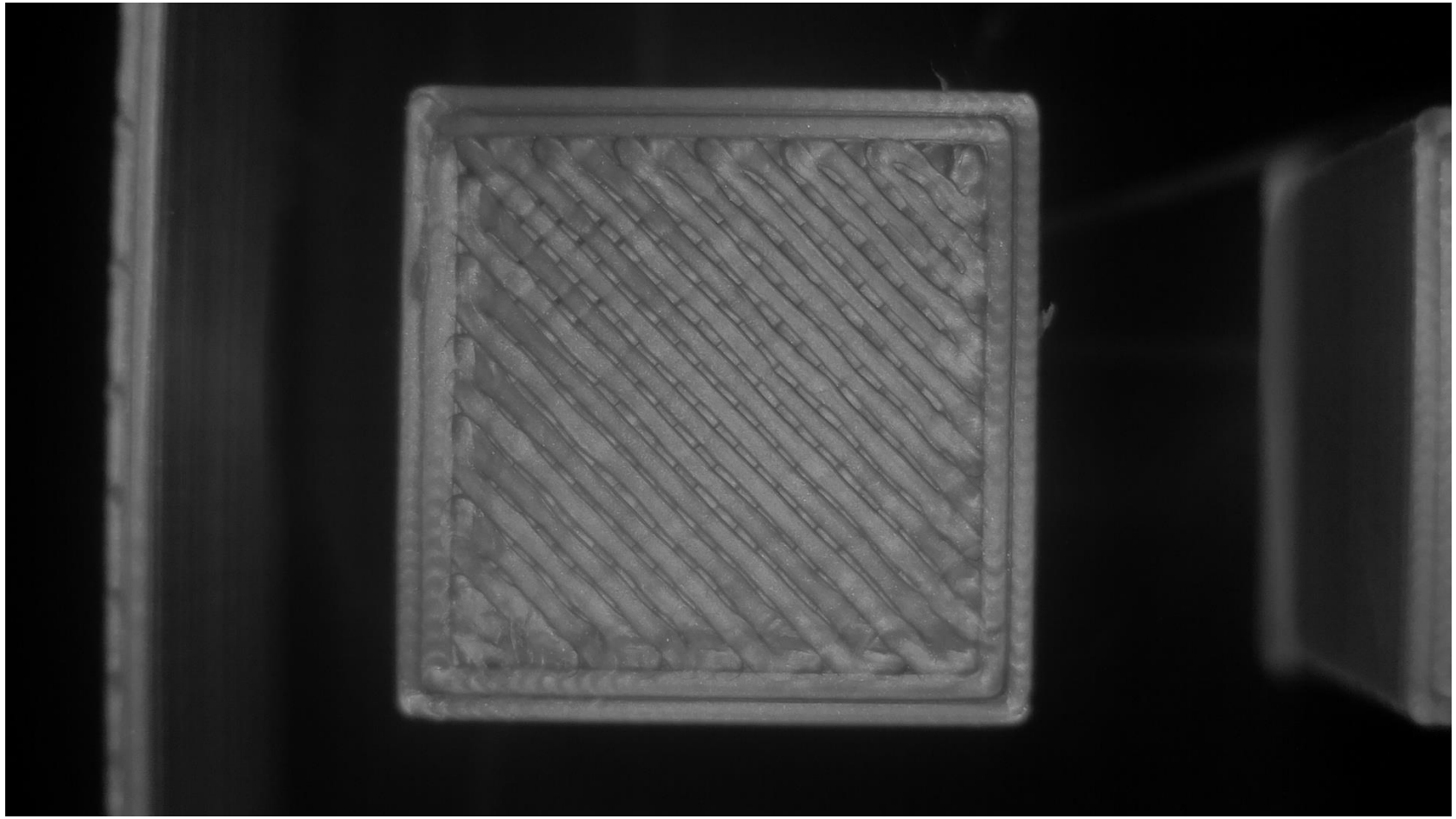
Printer Modification with Imager Assembly



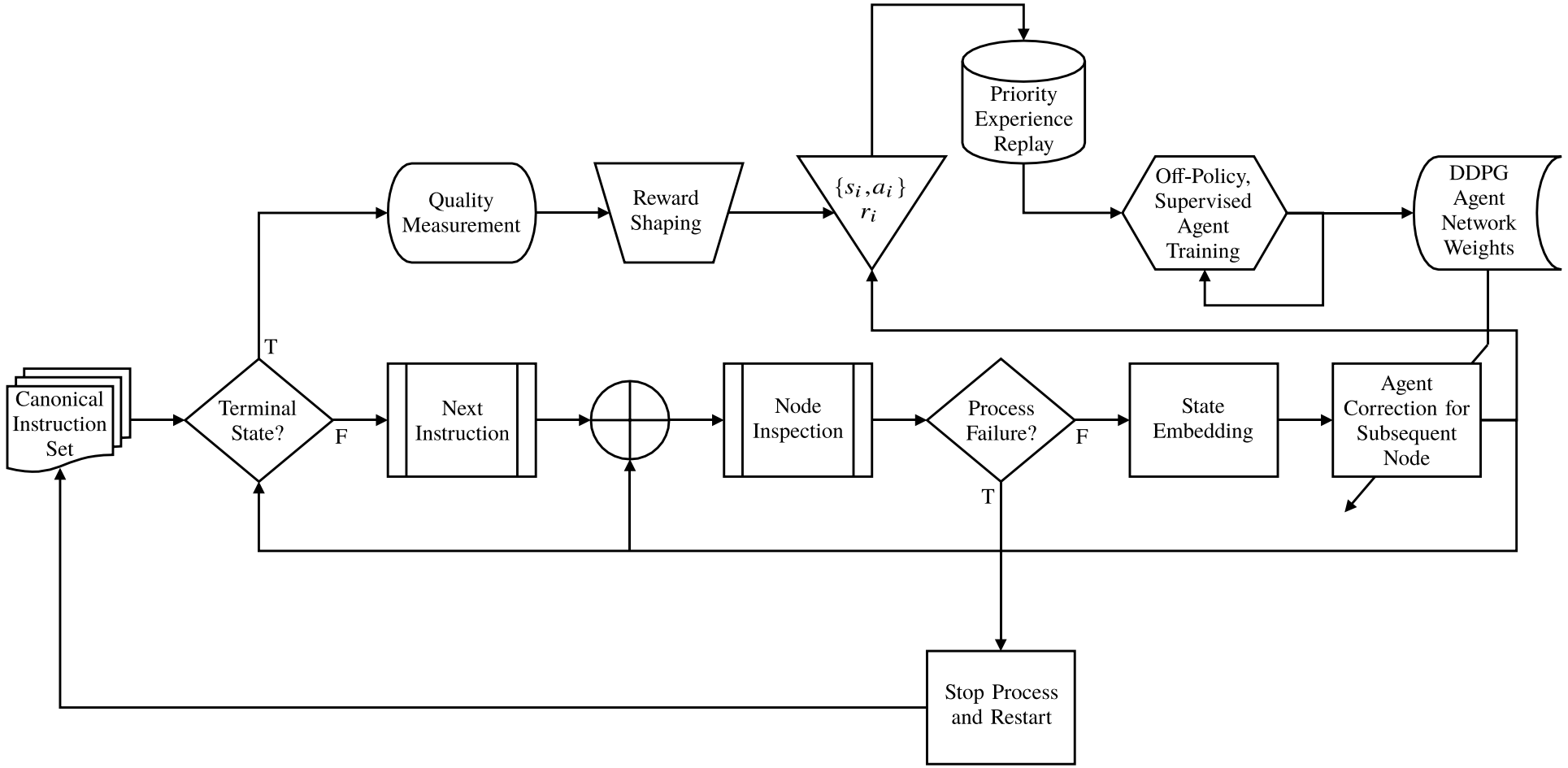




Example of Acquired Image



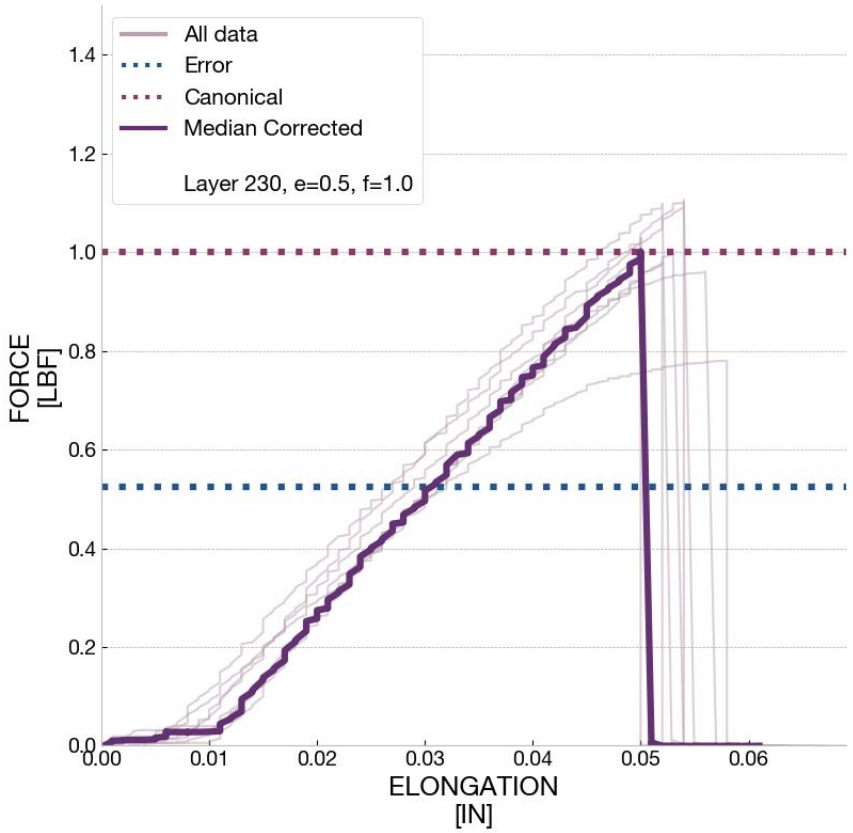
Feedback Loop Structure

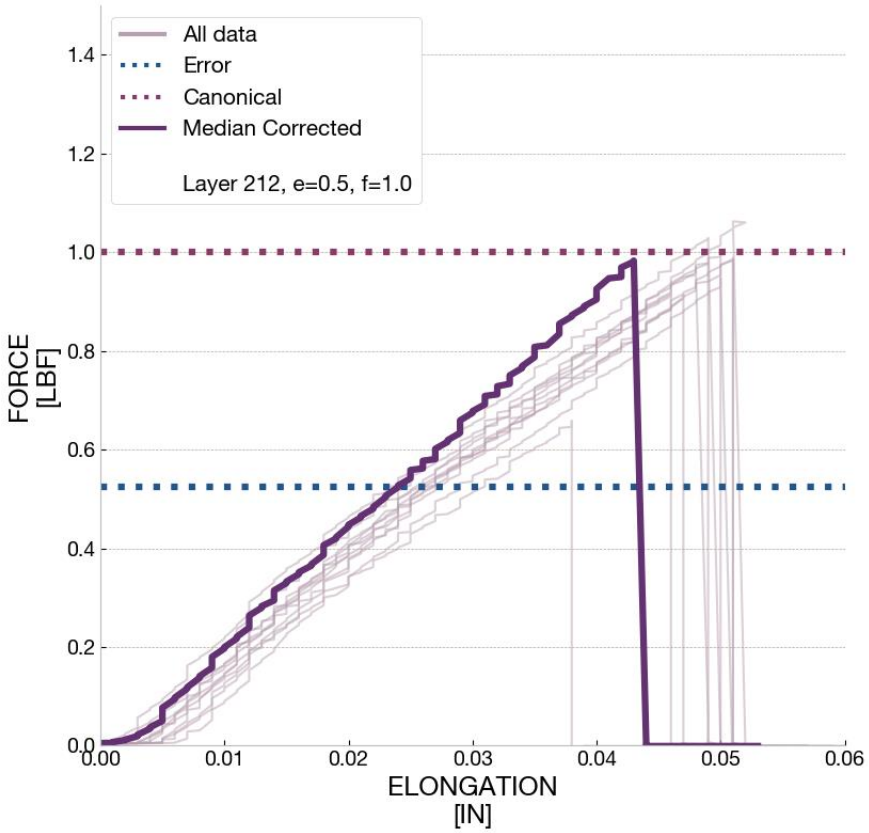


Discrete Correction Overview

- Discrete errors were artificially injected into the print by restricting the flow rate of plastic onto a particular layer.
- This error was classified by an image classifier trained with data from images of similarly restricted flow at certain degrees.
- The agent could adjust only *three* layers after the injected error.
- Baseline data was collected for the nominal (median) tensile strength and the median tensile strength of the specimen without correction.
- Finally, the resulting strength of the part after correction was measured.

Validation Results (Discrete Correction)

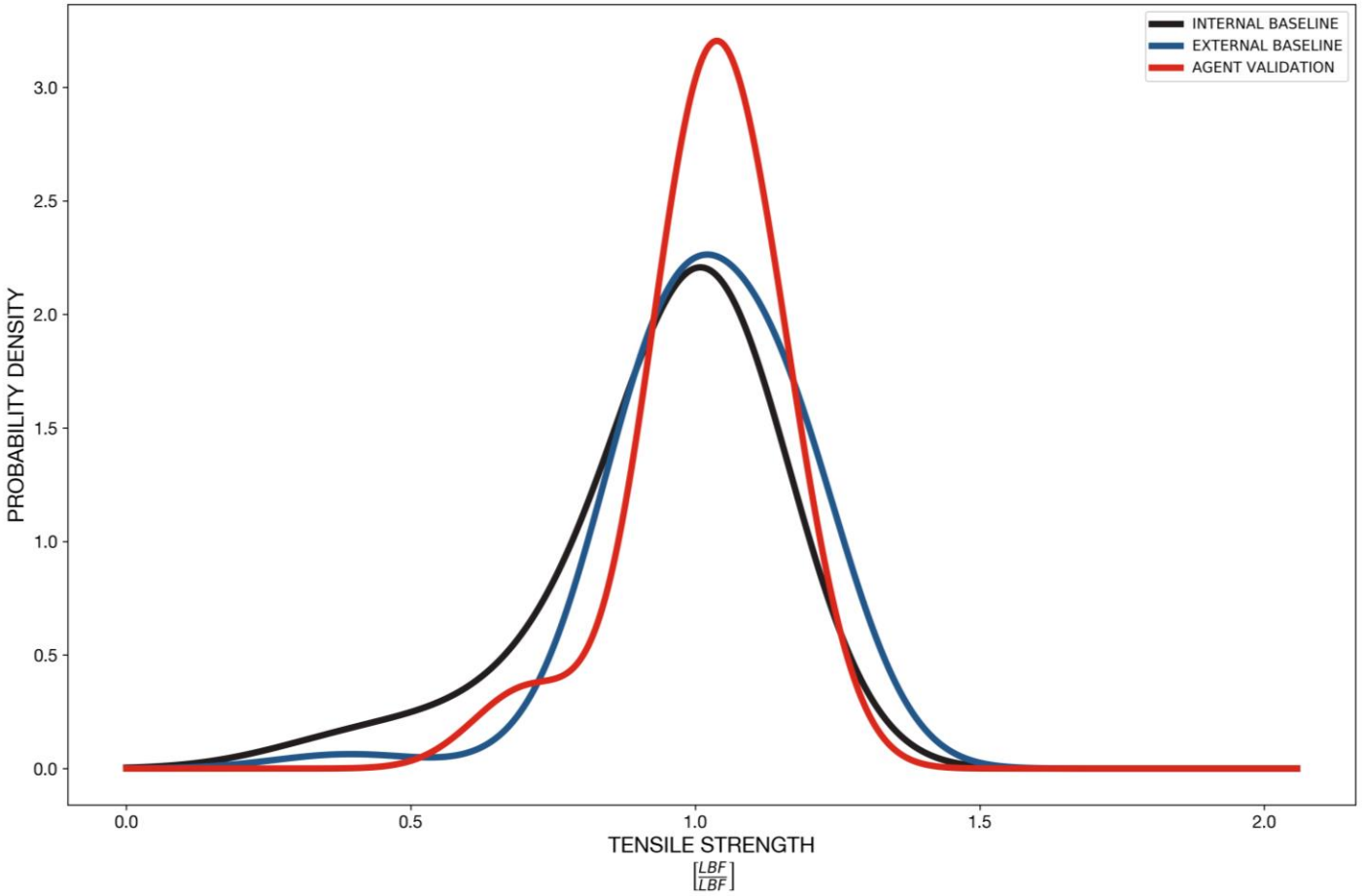




Continuous Correction Overview

- Discrete errors were no longer injected into the print, and instead the printer would print normally.
- Hundreds of thousands of images were captured of the printed layers, and the results were fed through a feature extraction encoder.
- The feature vectors were then used as the state for a reinforcement learning agent trained in an offline, off-policy fashion.
- A baseline was generated for a distribution of the expected tensile strength, and this distribution was confirmed by an outside measurement laboratory.
- The results from the corrective agent were compared against these.

Validation Results (Continuous Correction)



Insights and Conclusions

nanotronics

A Path Forward for the Next Industrial Revolution

- Image data is rich with information, and cameras are readily available in many form factors.
- While reinforcement learning in the context of physical systems is costly, an intelligent approach to dimensional reduction ensures the problem is tractable.
- We were able to construct an imager for an additive manufacturing system that provided a feedback loop to a reinforcement learning agent to infer corrective actions that:
    1. Classified and corrected discrete errors.
    2. Abstracted layer images to features for continual correction of minor errors.
- This work is applicable far beyond additive manufacturing, which was chosen for its widely transferrable principles. Industrial processes with defects or errors detectable through images can all be considered in this structure.

Additional Resources

Topics Covered

RL: Open AI – Spinning Up

https://spinningup.openai.com/en/latest/

Auto-Encoders

http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/

Additive Manufacturing

https://additivemanufacturing.com/basics/

2020 Embedded Vision Summit

"Imaging Systems for Applied Reinforcement Learning Control"

Details (Time, Date, etc.)