Jacob Voyles
March 15, 2023
Final Project Proposal

Deep Reinforcement Learning for Object Detection in Complex Environments with Active Annotation Selection

Problem:
The problem is to develop a deep reinforcement learning (DRL) model for robust and efficient object detection in complex with dynamic and occlude objects. Object detection is a crucial task in computer vision with various applications such as surveillance, robotics, and autonomous vehicles. Traditionally models would have to rely on handcrafted features and require a large amount of labeled data. This can be both expensive and time-consuming to collect. Deep learning-based vision models have shown impressive results in object detection but still suffer from needing large amounts of data, class imbalances, and localization errors. Thus I am proposing a reinforcement learning model for object detection that minimizes annotation costs which I believe will be a challenging yet promising research direction.

Relevant Prior Work:
1. Girshick, R. (2015). Fast R-CNN. arXiv:1504.08083.
2. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv:1506.01497.
3. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. arXiv:1506.02640.
4. Mnih, V., Kavukcuoglu, K., Silver, D., et al. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529-533.
5. Liu, W., Anguelov, D., Erhan, D., et al. (2016). SSD: Single Shot MultiBox Detector. arXiv:1512.02325.
6. Singh, B., Davis, L.S. (2018). SNIPER: Efficient Multi-Scale Training. arXiv:1805.09300.

Prior Experience:
I have some experience in creating deep learning and computer vision models, also having completed several object and action detection models using both Pytorch and TensorFlow. I have also worked with reinforcement learning techniques for decision-making in robotics and gaming applications.

Envisioned Approach:
1. Design a detector network
    a. Use an existing deep-learning model for object detection, like Faster R-CNN, as the backbone of the network. Use a region proposals network (RPN) to generate candidate object proposals. Train a CNN-based object classifier and bounding box regressor to redefine object proposals and predict object categories.
2. Develop a deep learning agent:

    a. Train an agent using a deep reinforcement learning algorithm, such as Deep Q-Netwrk (DQN), or an actor-critic method like A2C or PPO. From there, create a state representation that captures the relevant information about the image and the state of the network, and finally, define a reward function. The function would encourage the agent to select informative and diverse samples for annotation while minimizing annotation costs.
3. Incorporate active learning strategies.
    a. Create an estimation method for university within the detector network, like using entropy or variance for the predicted class probabilities.
    b. Design an exploration strategy that can balance exploring new samples for annotation and exploiting the current knowledge.
    c. Implementing a curriculum learning approach where the agent will select easier samples for annotation and gradually progress towards more difficult and uncertain samples as the model improves
4. Integrate the detector and reinforcement learning agent
    a. Create a training pipeline that will switch between updating the detector network and training the learning agent based on the detection performance and annotation cost
    b. Evaluate the performance of the detector network and agent on a validation set o monitor and adapt the learning strategies.

Code Starting Point:
I will start from the open-source implementation of the Faster R-CNN detector (https://github.com/rbgirshick/py-faster-rcnn) and modify it to integrate the reinforcement learning-based annotation selection agent.

Dataset:
I will use the COCO dataset (http://cocodataset.org/), which contains over 330k images with more than 2.5 million object instances labeled with bounding boxes and categories.

Architectures/Techniques:

1. Faster R-CNN for object detection.
2. Reinforcement learning-based methods, deep q learning.
3. Curriculum learning and active learning techniques to improve training efficiency and performance.

Open Questions/Unknowns:
1. How to balance the trade-off between detection performance and annotation cost?
2. How to design an effective reward function for the reinforcement learning agent?
3. How to generalize the model to new object categories and domains?

Computer Vision Components:
1. Object detection.

2. Bounding box regression.
3. Non-maximum suppression.

Deep Learning Components:

1. Convolutional neural networks.
2. Reinforcement learning.
3. Actor-critic methods.

Problem Difficulty:

This problem is challenging due to the need to balance the trade-off between detection performance and annotation cost and the complexity of integrating reinforcement learning with deep learning-based object detection. However, with proper design and optimization, developing an efficient and accurate deep reinforcement learning-based object detection model is feasible.