A-Team
Project Proposal Template

Please fill out this project proposal template before the deadline (**Thursday Feb 22, 2024 11:58 pm**). Submit 1 project proposal document per group. Keep your answers concise and clear!

1. **Team Members:**

   - Jaegook Alex Kim
   - McKay Moore
   - Sneha Masheshwari
   - Oscar Hasburn-Babich

2. **Problem Definition:** What are you trying to solve or do? Why is this a problem that is useful or interesting to study or experiment with? (2-3 bullet points).

   - **What:**

     In warehouses, certain utilitarian objects or physical “assets” –e.g. pallets, empty pallet rack space, industrial carts, etc.— aren’t usually inventoried by default, nor treated as inventory targets by existing enterprise resource planning (ERP) systems, at non-trivial impact to stakeholders. The effort of human auditors performing routine elicitation tours for the purpose of inventory is costly, time consuming, and error prone –even in absence of human opportunity cost consideration.

     Our goal is to relieve the human effort and mechanize object location and inventorying tasks at faster overall speed and lower cost. We seek to leverage a lightweight, real-time detection architecture, in combination with edge inference technologies like NVIDIA Jetson boards to deliver our solution.

     As a potential alternative, we notice similar needs in secure public bike parking. In pre-planned environments, bike parking spaces often have detection systems to help arriving cyclists find an open space. Aerial object detection assists with accurate growth planning via longitudinal data collection of bike traffic and parking usage, and embedded detection with available open space identification in real-time, via retrofit on existing parking structures.

     We have identified a few candidate target objects or “assets” (e.g. pallets, etc); we are looking forward to receiving your feedback and refining scope and selection under your guidance.

   - **Why?**

     In Warehouses:

- Even before COVID, businesses experienced pallet stock shrinkage at a total direct cost of $25+/pallet, and indirect cost in the aggregate thousands of $USD per year from distribution misses. From informal interviews, we learned that the problem significantly worsened during COVID, compounded by prohibitive lead times and increased prices. Pallets became a hot commodity.

- Storage space in low-automation warehouses is vastly estimated from human visuals, frequently leading to under and over-utilization periods of available space at non-trivial operational impact.

In cities, on campuses:

- Proliferation of bikes as transportation means in main European cities and University campuses increased secure bike parking planning needs, and proper provisions for placement and sizing. Oftentimes, secure parking space is oversubscribed, and available parking remains elusive during peak hours; the absence of longitudinal data usage makes difficult the justification for additional space provisions, for convenience and asset protection –bike parking structures are inexpensive; the impact of lack thereof isn't.

- Related cost and losses are unknown to us as of this writing, but we estimate non-trivial impact on user experience and derived economic activity, even in absence of theft.

3. **Project Idea:** What do you plan on accomplishing in your project scope? How will your idea solve the problem proposed in your problem definition? (2-3 bullet points)

- Deliver a terrestrial or aerial vehicle-mountable, cost effective, accurate, small-footprint, real-time object detection solution capable of procuring location and count of an essential asset from the candidate list (e.g. pallets, empty rack space, bikes, empty bike parking space), to replace existing manual inventorying needs.

- Architectures like YOLO's and technologies like NVIDIA Jetson enable object detection at the edge in real-time; when mounted on a vehicle, at a fraction of human-based effort, in time and overall cost. A straightforward solution comprises a camera, a Jetson board, a properly trained model, and ancillary assets.

- Rather than:

    - Embarking on time-consuming facility tours, asset managers can rely on forklift-mounted detection for inventorying, and even receive alerts when notable events occur: *"4 pallets were loaded on outbound trucks at 10:35 AM, 20 were received."* Stakeholders can have access to accurate data in real-time and wholistically overview the flow of how assets move around the facility and improve processes.

    - Wandering for minutes in crowded parking spaces (picture yourself outside the library at peak hours on a school day, or a busy train station at rush hour), electronic displays can guide users as to which racks, isles, or regions have open parking spaces. The object detection system will inform the display counts.

4. **Related Works:** Briefly describe what work has been done in this area and how they currently approach the problem. List any research papers, existing codebases, or miscellaneous articles/websites that you found that relate to your work. (minimum 5 references; at least 3 should be research papers)

    - Similar detection in near real-time and real-time of assets has been done, mostly without modular edge productization as a main goal, using earlier YOLO versions with benchmarking against SSD and Faster R-CNN architectures, and consuming publicly available datasets –augmenting data in a few cases. Technologies involved include TensorFlow and PyTorch.

    - As this is a timeboxed course project and not a research initiative to produce a whitepaper for a journal, we believe our choice is sensible and unique in its intended goal, including its focus on post-training optimization for edge inference from a moving vehicle, and provisions for offline batch download and wireless transmission of inference results –location and inventoried data. Starting with a latest YOLO version and focusing on maintaining robust mid-range distance detection accuracy, we can meet deadlines with a quality result, and benchmark ours against prior attempts utilizing earlier architectures at static and slower speed capture points.

- Object detection using YOLO: challenges, architectural successors, datasets and applications
    - This paper goes in-depth on analyzing YOLO architectures and discusses the evolution of object detection from traditional methods to deep learning approaches.

- A Comparison of Deep Learning Models for Pallet Detection in Industrial Warehouses
    - This paper uses three state-of-the-art CNNs: Faster R-CNN, SSD, and YOLOv4 to detect pallets in warehouses to automate the task for detecting pallet storage and retrieval.

- A systematic review on computer vision-based parking lot management applied on public datasets
    - This paper used publicly available image datasets to craft and test computer vision-based methods for parking lot management approaches.

- A deep learning approach to real-time parking occupany prediction in transportation networks incorporating multiple spatio-temporal data sources
    - This paper creates a Graph-Convolutional Neural Network to extract spatial relations of traffic flow in large-scale networks to predict parking occupancy.

- Automatic parking space detection system based on improved YOLO algorithm
    - This paper develops a parking space detection system using YOLOv3 with a variety of datasets from different occasions, time periods, and lighting environments.

5. **Datasets:** How will you benchmark your approach or theory? Describe and link each dataset you will use in your project, and explain why you selected the dataset. (min 1 dataset)

- Modern object detection in real-time is well-understood and has been successfully applied; we only intend to author, if necessary, minimal architectural improvements over our detection architecture of choice, and are planning to test progress and model accuracy against physical assets in their corresponding environments. We have brokered access to select warehouse space (at a former employer of a group member) in the case of warehouse assets and have access to campuses in the case of bikes and bike parking spaces.

- As for data, choice is straightforward in our case, and we plan on leveraging:

    - Corresponding, publicly available datasets:
        - Warehouse items in the case of warehouse assets, as linked above.
        - Bikes and bike racks in the case of bikes and bike parking space.

    - Directly sourced high-quality images:
        - From colleagues –in Supply Chain in the case of warehouse assets, etc.
        - Our own pictures –of pallets, racks, bikes, bike racks, whatever the choice.

    - NVIDIA TAO for data augmentation
        - https://developer.nvidia.com/tao-toolkit

- Selected Datasets (listing the more relevant, we have compiled a larger set):

    - Warehouse Objects Datasets
    - Pallets Dataset
    - Bike Rack Image Dataset ~ 200 images
    - Bike Image Dataset ~ 400 images

6. **Proposed Experiments:** What experiments will your team perform in your project? Roughly describe what ideas you will try. (2-3 bullet points)

- Sourced, collected, and augmented data will be visually reviewed by the team.

- Local Testing

We have immediate access to a limited but sufficient set of assets outside of their native environments, and to portable computing equipment which we intent to use for initial testing and fine-tuning. We will iterate over the development cycle –data exploration, model selection, model training, model fine-tuning, deployment, testing—and do initial experimentation at home.

- Site Testing

  We have access to select, fitting, native environments such as low-automation warehouses, and to school campuses for post-initial testing.

  In our case, the testing approach is straightforward: we will select 2-3 camera models and narrow down to 1, we have performed successful initial inference testing on a Jeston Xavier NX using an Intel RealSense and a pre-trained model on household items, we will focus on recent YOLO architectures, consume our target training data, and run inference from our hands for testing. We will collect inference results, analyze, fine-tune, rinse, repeat. If doing warehouse asset detection (e.g. pallets), site testing involves mounting the solution on a forklift and collecting results after a warehouse tour is completed.

7. **Compute Resources:** What compute resources (GPUs, CPUs, TPUs) does your team plan to use? The type of compute resources can limit the scope of your project, so we want teams to pick appropriate projects for the resources they have accessible.

- We made the selection of our topic and project based on affinity and compute resources directly available to us. We believe we have just enough resources at our disposal. They are as follows:

  - 1x MacBook Pro 14 M3 Max 128GB RAM 8TB SSD
  - 1x Razer Blade 15 RTX 3080Ti 16GB VRAM, 64GB RAM 4TB SSD
  - 1x Razer Blade 16 RTX 4090 16GB VRAM, 64GB RAM 4TB SSD
  - 1x Jetson Xavier NX Developer Kit
  - 1x Server Rack with 4x NVIDIA A100 GPUs

Final Project Details

Requirements

As part of CS 6476, you will be completing a final project, which is an open-ended assignment with the goal of applying skills and knowledge from the class to real-world applications or research problems. Students are expected to be in groups of 4 students. After each group submits a project proposal, you will be assigned a TA that will serve as a mentor. The final project deliverable will be a github website (e.g template) that will thoroughly describe the scope of your project, experiments, and results. The final project will be graded on these 5 main components:

- **Introduction/Problem Definition:** Provide a brief introduction to your project topic and describe why it's an interesting topic to investigate. This is where you want to describe the problem itself and the motivation behind tackling it.

- **Related Works:** Describe related works in your problem space (research papers, libraries/tools, etc.) for existing solutions for this problem or adjacent areas. Make sure to cite papers you reference!

- **Methods/Approach:** Indicate algorithms, methodologies, or approaches you used to craft your solution. What was the reasoning or intuition for trying each methodology/ algorithm. What does the overall pipeline look like and the details behind each component? Make sure to establish any terminology or notation you will continue to use in this section.

- **Experiments / Results:** Describe what you tried and what datasets were used. We aren't expecting you to beat state of the art, but we are interested in you describing what worked or didn't work and to give reasoning as to why you believe so. Compare your approach against baselines (either previously established or you established) in this section.

- **Conclusion/Discussion:** Summarize your main findings or insights, and discuss what you would do to further this work? What unexpected problems did you notice in your current setup and how would you mitigate them in a future iteration?