Instructions for Final Project Report

1. Write an executive summary which is no more than 250 words to describe your overall project, the approach, the highlights of your discoveries/key findings, and your overall result. This is a short abstract in effect that summarizes the entire report.

2. Introduction – approximately half page on describing the problem, and what your objectives are, and the techniques you considered for solving the problem.

3. Technical Description – at most 2 page on the technical approach, describe the architecture, and the algorithm(s) in sufficient detail. Describe any innovations and contributions you made to the algorithms. Describe the end to end process for detection and/or classification (you are to use separate detection and classification stages for options 1 and 2). The detector should nominate a region of interest, and the classifier must provide the class label. Use figures as appropriate to illustrate the approach. You may have tried several approaches, but in this section limit it to your final solution/algorithms.

4. Data set – approximately half page description of the data (including training and testing sets). Use figures as needed to show examples of the data

5. Results – At most 2 pages to describe your results (focusing on detection and classification for options 1 and 2, and just classification for option 3). Describe the metrics to be used, and present the results obtained using these metrics (e.g. ROC curves, confusion matrices, location accuracy, detection and recognition performance, IoU, map, etc). Use figures and tables as necessary to show quantitative results.

6. Conclusions – at most 1 page on your key conclusions, the insights into the problem (i.e. what worked and what didn’t), and the key limitations/advantages of your approach. Remember it is valuable to know the bottlenecks and challenges, and what you did to overcome them. List your discoveries along the way.

7. Any appropriate references

8.Submit the source code and executable that can be used for testing, along with a sample images on which the code can be run to observe the results. If you use video, then submit a video sequence on which the code can be run. The results should be clear (such as detections and class labels annotated on the image, and printed on the screen).

Report Status

1 – executive summary: draft complete

2 – introduction: short, can be written whenever

3 – technical description: long, can be written whenever, but may need updates

4 – data set: short, and can easily be written whenever

5 – results: long, requires data, and only the classifier alone can be written

6 – conclusions: short/medium. Heavily depends on final performance.

7 – appropriate references: -

8 – source code submission, sample images for code input: -

1. Write an **executive summary** which is no more than 250 words to describe your overall project, the approach, the highlights of your discoveries/key findings, and your overall result. This is a short abstract in effect that summarizes the entire report.

Using the FLIR dataset for infrared imagery in urban environments, we sought to create models to process the data and produce appropriate bounding boxes and classification labels for objects within the given scene. To accomplish this, we implemented a two-model solution, where the first model is trained to detect and determine bounding boxes for objects of interest, the Detector, and the second model is trained to classify the bounded images that may occur, the Classifier. We found that our designed model for the Classifier performed to desired metrics. The Detector’s performance, however, was insufficient in reaching the desired goals and capabilities, suggesting that our implementation may have been theoretically achievable, but constrained by the performance of that model. Research into the setting showed a common alternative architecture that combines both the Detector and the Classifier into a single model; the lack of success in our designed architecture serves to inform our conclusion that alternative approaches may show more success.

2. **Introduction – approximately half page** on describing the problem, and what your objectives are, and the techniques you considered for solving the problem.

Short. No results or conclusions, so a quick summary of the way we tackled the problem. Can be written without consideration of pending work.

Problem and Objectives:

* Detecting and classifying objects – cars and people, specifically – from infrared data

Techniques Considered:

* Combined model
  + Train classifier and detector as part of one model
  + Training of each part is interdependent
* Split model
  + Classifier model and detector model
  + Each can be trained separately
  + Performance is measured separately
  + Output of detector feeds into the input of the classifier
* YOLO?
* Grid system/whatever that was in the PowerPoint about object detection?
* (pretty sure there’s more)

3. **Technical Description – at most 2 page** on the technical approach, describe the architecture, and the algorithm(s) in sufficient detail. Describe any innovations and contributions you made to the algorithms. Describe the end-to-end process for detection and/or classification (you are to use separate detection and classification stages for options 1 and 2). The detector should nominate a region of interest, and the classifier must provide the class label. Use figures as appropriate to illustrate the approach. You may have tried several approaches, but in this section limit it to your final solution/algorithms.

Long and can mostly put it together/write it, though may need updates if we make adjustments. End-to-end detail of what we did.

Architecture:

* Detector model
  + batchNorm
  + “features”
    - Resnet18
    - Conv2d
  + ReLU
  + F.adaptive\_avg\_pool2d
  + reshape/view
  + bb
    - linear
    - ReLU
    - Linear
* Detector loss function
  + SmoothL1Loss
  + MSELoss
* Classifier model
  + batchNorm2d
  + “features”
    - Resnet18
    - conv2d?
  + relu
  + F.adaptive\_avg\_pool2d
  + reshape/view
  + classifier
    - linear
    - relu
    - linear
    - softmax
* Classifier loss function
  + CrossEntropyLoss
* (all that can be more accurately summarized by looking at the output from torchsummary)
* Refined version of the powerpoint picture that shows our dataflow between the model stages
* Show how it should’ve worked, and how the way we drew it up made sense

4. **Data set – approximately half page** description of the data (including training and testing sets). Use figures as needed to show examples of the data

Short and can put this together whenever.

Woohoo pictures and talking about the data briefly.

* Pictures directly from the data

5. **Results** – **At most 2 pages** to describe your results (focusing on detection and classification for options 1 and 2, and just classification for option 3). Describe the metrics to be used, and present the results obtained using these metrics (e.g. ROC curves, confusion matrices, location accuracy, detection and recognition performance, IoU, map, etc). Use figures and tables as necessary to show quantitative results.

Long and requires data. Should wait until we’ve settled on our data, though info about the classifier can be organized and put here.

* Detection went terribly.
* Classification rocked.
  + 96% accuracy on classification, booyah.
* This will be taken up by a lot of numbers, graphs, etc.

6. **Conclusions** – **at most 1 page** on your key conclusions, the insights into the problem (i.e. what worked and what didn’t), and the key limitations/advantages of your approach. Remember it is valuable to know the bottlenecks and challenges, and what you did to overcome them. List your discoveries along the way.

Short/medium. Mostly should wait until we’ve settled on our performance.

Points to talk about

* Classifier training worked great
  + Leveraged training ideas from previous homework
* Creating the custom ThermalCocoDataset class
  + A lot of custom work
* Advantage of our approach: ability to perform separate trainings and tunings
* Advantage of our approach: traceable behaviors by the models
* Disadvantage of our approach: need to train twice
* Disadvantage of our approach: an image has a longer dataflow for real-time processing
* Disadvantage of our approach: detector didn’t work
  + Suggest using more developed patterns for detection, ie powerpoint from class
* Challenge: multi-OS development, overcame the numWorkers issue
* Challenge: multi-OS development, unified environment using conda environments
  + Added bonus: more easily repeatable by others

7. Any appropriate **references**

8.**Submit the source code and executable that can be used for testing, along with a sample images on which the code can be run to observe the results**. If you use video, then submit a video sequence on which the code can be run. The results should be clear (such as detections and class labels annotated on the image, and printed on the screen).