Module 1 Project: Computer Vision Application

Machine Learning

Fall 2021

Computer vision and image classification tools are rapidly reshaping our lives. Machine learning approaches have driven recent progress in an array of technologies that have the potential to realize huge positive impacts on our world. These tools have become increasingly easy to apply to masses of existing data, however, these tools and their real-world application do not always have the results that their creators intend. To achieve better outcomes, it is important to deeply consider the potential implications and limitations throughout the application design and implementation process. This project aims to help you to begin to develop the skills to understand, implement, and critically evaluate machine learning systems.

This document is quite long and contains links to other information and resources. The table of contents below should help you nagivate to various sections.

Contents

1   Goal-Setting and Customization

In the words of one of your course assistants, Jerry, your instructors define a good project as one that successfully uses a neural net on a dataset and demonstrates iteration (you try the neural net, you make improvements, you try again). If you wish to go deeper, that's great!

However, we also (informally) ask you to consider your own learning goals. Is there a specific skill you want to practice? Is there a way of learning (like, asking for help more, trying code without looking at examples first, etc) that you want to practice? We ask you to think about these, and ask your partner about their learning goals. You can customize this project to support your own learning, and we are happy to help you shape the project to support your goals and challenge yourself.

2   Project Description

You will be working in teams of two. In this project, you will contemplate a potential application for computer vision (machine learning of images) and you will implement a machine learning algorithm on a data set in support of this application. As part of this project, you will:

- Document the important considerations for your application (e.g., what data are available for training, how well would the algorithm need to work to create value, how could the algorithm and application be tested (beyond the testing you choose to implement), what are the implications of this application in the world, what are the stakeholders, what are the risks).

- Build and iterate on a computer vision model as a step toward this application. We don't expect you to build the entire application, just to work toward implementing an early version of an algorithm that could be used for this application. It's likely that you will use a convolutional neural network, but not required. Please talk with us if you plan on using a different type of model. You will build and test multiple versions of your model. You can use pre-built libraries like pytorch.

- Visualize key aspects of your data/model. Include at least two visualizations in your report with clear labeling and a discussion of their meaning (though you'll likely want more than two visualizations). You'll probably find it valuable to visualize many aspects of the data and model as you work. This is helpful in sanity checking and can give good insight into how the model is working.

- Test and evaluate your model. This will likely include accuracy on training and test sets. Depending on your application, you may also want to collect your own photos, find other images online, and/or manipulate images from your original test set (adding noise, shifting/flipping images, etc). You will also evaluate the effectiveness of this model for your specific application and describe the limitations of your current model and data.

- Document your final analysis pipeline for transparency. (A simplified version that does not need to include every parameter, visualization, or tweak that you tried.)

3   Timeline

The project will officially launch on October 1, 2021 and end on October 22, 2021 (report may be due before). We will have several interim deadlines and some other activities happening during class that may require preparation. We may choose to adjust deadlines based on class progress and feedback. We will keep a tentative schedule in this spreadsheet and use Canvas to share interim deadlines.

4   Deliverables & Assessment

4.1   Deliverables

1. A clean report in notebook (.ipynb) format with thoughtful explanations of the considerations for your computer vision application and important highlights of your analysis. This should include in-line code and figures. This should NOT include every analysis and figure that you generated. Please choose three interesting things that you investigated and share your code and figures for these. You can mention other things that you investigated in the text and include the code in your repository. This is about quality, not quantity.

2. A pdf of your clean notebook with all of the cells run and graphs appearing clearly.

3. Your self-assessment of constructive engagement (letter grade and explanation with descriptions of your actions and/or pointers to evidence in your work).

   There are rubrics in the appendices in this document. The intention of these rubrics is to create shared expectations about the project, while also giving you lots of latitute to explore and emphasize your own learning.

4.2   Assessment

Assessment for this project will be evenly split between **constructive engagagement**, which is self-assessed by you, and **quality of project work**, which will be assessed by faculty.

50% Constructive engagement (individual): You will self-assess based on our shared rubric and include an explanation of your self-assessment. This aspect of your grade will be largely based on your self-assessment, but we may adjust your constructive engagement based on your explanation and our observations.

50% Project work and report (team): We will assess and provide feedback on your final report. This should be in the form of a Jupyter notebook. This notebook should not be an extensive record of all the work you did, rather a polished report that explains the work you did with accompanying code.

We want to acknowledge that we are all coming to this course with varying experience with machine learning. This is great, and we're actively trying to help you all have positive learning experiences. We intend for the course not to require prior experience with machine learning to participate, learn new things, and feasibly receive a high grade. With this in mind, we want to be explicit about how we anticipate this playing out in assessment.

With the project work and report, we will assess with a mental model of a team beginning the course with no real machine learning experience. The key aspects of this assessment are described in the rubric. In practice, this means that everyone could receive an A for this part, and it will be more challenging for some people than others.

With constructive engagement, we are asking you to self-assess. One aspect of constructive engagement involves challenging yourself. We're trying this strategy to avoid the situation where we try to estimate your previous experience and base our expectations around these assumptions. This relies on an honest self-assessment of your own constructive engagement.

4.3   Class-owned editable resources file

We will continue to use Discord and office hours to answer questions. However, we also know that a great deal of learning happens outside of those places. This shared, class-owned, editable document is intended to serve as a communal set of resources for troubleshooting as we figure things out together. Anyone can add to it, so as you run into problems (you will) and figure out solutions (sometimes on your own, sometimes with help), please add to this document to help others. Learning is not a zero-sum game.

We also want to note that the field of machine learning is constantly changing, as are things like libraries and toolboxes. It's likely that you'll run into something that worked 2 years ago and does not work now. Please help us find and fix these things!

5   Appendix A: Some questions to consider

These questions are taken from this webpage.

- What do you want your model to be able to do?

- How can you imagine your model (or an extension of it) being used in the real world? Feel free to get creative.

- If those ideas came true, who might they affect? In what ways?

- What measures would your model's real-world implementers need to take to ensure its effects live up to your intentions?

- What pitfalls might your model fall into, and what could you quantitatively measure to avoid those?

- Why was the dataset you used to train your model created?

- In what other ways could the same data be used? How do you feel about those possibilities?

- How was that dataset assembled? From where was the data sourced? Who or what labeled it? If there are any elements of this process you think were either particularly well done or problematic, how so?

- If your dataset contains information about people, to what degree did those people have agency over their inclusion? Do you feel that matters in this case? Why or why not?

- Skimming through your dataset, does anything stand out to you about representation in its contents?

- Do you feel the potential use cases for this dataset justify it being created and published? Why or why not?

6   Appendix B: Computer Vision Datasets

6.1   TensorFlow built-in

There are numerous datasets for Image Classification on the Tensorflow datasets documentation page. We strongly suggest choosing one from the Image Classification category (see tab on left) instead of the other categories, unless you have substantial prior machine learning experience.

- Cats and Dogs: 25,000 labeled images of cats and dogs

- CelebA: 202,599 total face images of 10,177 identities w/ facial landmarks, aligned & cropped images, bounding boxes, and binary attribute labels

- CIFAR-10: 60,000 labeled 32x32 color images of 10 classes of objects

- CIFAR-100: 60,000 labeled 32x32 color images of 100 fine classes, also grouped into 20 course superclasses

- CIFAR-10-C: Images from CIFAR-10 manipulated using 15 common corruptions at 5 levels of severity each

- Colorectal Histology: 5,000 labeled 150x150 color histological images of 8 tissue types of human colorectal cancer

- CBIS-DDSM: 2,620 labeled scanned film mammography studies of normal, benign, and malignant cases of human breast cancer

- Diabetic Retinopathy Detection: 88,712 labeled images of eyes with diabetic retinopathy at none, mild, moderate, severe, and proliferative levels of severity

- EMNIST: 800,000 labeled 28x28 grayscale images of handwritten digits and letters (uppercase and lowercase) that expand the MNIST dataset

- Fashion-MNIST: 70,000 labeled 28x28 grayscale images of 10 classes of clothing articles

- Horses or Humans: 1,283 labeled 300x300 color images of cgi humans and horses

- ImageNet 2012: 1,331,167 labeled color images of tens of thousands of classes of nouns in the WordNet hierarchy

- ImageNet-C 2012: Images from ImageNet 2012 manipulated using 12 common corruptions at 5 levels of severity each

- KMNIST: 70,000 labeled 28x28 grayscale images of 10 classes of handwritten Japanese Hiragana characters

- LSUN: 708,564 labeled large-scale color images of 10 classes of scenes/settings

- MNIST: 70,000 labeled 28x28 grayscale images of 10 classes of handwritten digits

- MNIST-C: Images from MNIST manipulated using 15 common corruptions at 5 levels of severity each

- Omniglot: 38,300 labeled grayscale images of 1,623 classes of handwritten characters from 50 alphabets w/ stroke data in [x,y,t] coordinates where time (t) is in milliseconds

- Oxford Flowers: 8,189 labeled color images of 102 classes of flowers

- Oxford-IIIT Pet: 7,349 labeled color images of 37 classes of pet breeds w/ tight bounding boxes around the heads and pixel-level foreground-background-boundary segmentation

- PatchCamelyon: 327,680 labeled 96x96 color images of histopathologic lymph node scans where metastatic tissue is either present or absent

- PetFinder: 72,776 color images of dogs and cats w/ many attribute labels

- Quick, Draw! Bitmap: 50,426,266 labeled 28x28 grayscale images of 345 classes of drawn objects contributed by Quick, Draw! players

- NWPU-RESISC45: 31,500 labeled 256x256 color images of 45 classes of scenes/settings

- Rock Paper Scissors: 2,892 labeled 300x300 color images of cgi hands in rock, paper, and scissors gestures

- small NORB: 48,600 labeled 96x96 color images of 5 classes of toys

- Sun397: 108,753 labeled color images of 397 classes of scene/settings

- SVHN-cropped: 600,000 labeled 32x32 color images of digits of house numbers (addresses) obtained from Google Street View

- TF Flowers: 3,670 labeled color images of 5 classes of flowers

- UC Merced Land Use: 2,100 labeled 256x256 color aerial images of 21 classes of land uses collected from USGS National Map Urban Area Imagery

6.2   Other datasets

- Bark-101: 2,594 labeled color images of 101 classes of tree bark

- Caltech 101: 9,146 labeled color images of 101 classes of objects

- Caltech 256: 30,607 labeled color images of 256 classes of objects

- CCPD: 300,000 labeled color images of license plates w/ bounding boxes, license plate numbers, and several dimension labels

- CINIC-10: A drop-in replacement for CIFAR-10 with 270,000 images collected by downsampling images from ImageNet

- DFW: 11,157 labeled color images of 1000 subjects' faces, including examples of that subject attempting to obfuscate their face and of other individuals attempting to impersonate that subject

- FERET: 14,126 labeled 384x286 color images of 1,119 faces

- Food-475: 247,636 labeled color images of 475 classes of food

- Food-524: 247,636 labeled color images of 524 classes of food

- HistAerial: 4,900,000 labeled grayscale aerial images of 7 classes of ground use

- IMDb: 460,723 color images of celebrities w/ bounding boxes around faces and attribute labels including DOB, name, binary gender (literally)

- IP102: 75,000 labeled color images of 102 classes of insects

- LAG: 11,760 labeled color images of suspicious or negative glaucoma samples

- LFW: 13,233 labeled 250x250 color images of 5,749 faces

- LFW-a: Images from LFW that have been aligned using a commercial face alignment software

- LFWcrop: Images from LFW that have been cropped to only include the contents of facial bounding boxes

- ML-Images: 17,698,491 labeled color images of 11,166 classes of objects

- Multi-PIE: 750,000 labeled color images of 337 faces

- PIE: 41,368 labeled color images of 68 faces

- QMNIST: An extension of the MNIST data set to include 50,000 additional test images

- SCface: 4,160 labeled color images of 130 faces taken from 6 security cameras of varying quality in either visible light or infrared mode

- SoF: 42,592 labeled color images of 112 faces (exclusively glasses-wearing)

- Stanford Dogs: 20,580 labeled color images of 120 breeds of dogs w/ bounding boxes

- VERI-Wild: 416,314 labeled color images of 40,671 vehicles

7   Appendix C: Rubrics

7.1   Constructive engagement

You will self-assess your constructive engagement using the rubric here. This rubric was designed in collaboration with the students from the first iteration of this course. Instead of re-inventing the wheel, we will discuss this rubric as a class and make revisions as appropriate.

7.2   Jupyter Notebook

Please submit both a pdf of your Juptyer Notebook (with the code executed) and an executable Jupyter Notebook to Canvas.

We suggest you work in the Google CoLab environment (like you have for the assignments). When you are ready to submit, please download your notebook as an ipynb file, and submit that to Canvas.

Your project will be graded on the following aspects. **General**

1. The notebook is well-organized.

2. The notebook balances code and text well.

3. The notebook runs without errors in the Google CoLab environment.

4. The notebook is free of typos.

5. The notebook demonstrates good coding practices, including but not limited to docstrings, appropriate variable names, and comments as appropriate.

6. The code demonstrates a significant amount of effort (roughly 3 assignments' worth), made clear by careful consideration of the data, model, and implications for the model you developed.

Motivation

7. It is clear what data you used.

8. The application of your algorithm is clear, and well-motivated.

9. The notebook explains how well the algorithm would need to work to provide value

10. The notebook explains how the algorithm is to be evaluated (how do you know it is working well?)

11. The notebook explains the implications of this approach (including risks and stakeholders)

Implementation

12. The notebook explains the dataset and the features, in text and/or graphics.

13. The notebook contains a computer vision model.

14. The computer vision model is implemented correctly.

15. There is evidence that iteration has occurred in the model (this can be in the text).

Conclusion

16. The notebook accurately describes the effectiveness of the model in text.

17. The notebook describes the limitations of the model.

18. The code generates at least two visualizations of the data.

19. The visualizations have clear labels and connections to the text.