Scope

In this project you will attempt to build a working vision system.

Make your project interesting and fun and keep it simple. You will only have about four-five weeks to carry it out. It is far better to do something simple that works, than to work on something ambitious but are unable to carry out experiments.

Set for yourself two goals:

1. *minimal goal*: you should be able to make it to this point even if you get unlucky;
2. *ambitious goal*: something to aspire to if all goes swimmingly, or if you decide to continue your project beyond the class.

The deliverables from the project are:

1. Proposal presentation (2' + 3' Q&A)
2. Final presentation (5' + 3' Q&A)
3. Project final report (8 pages)

Questions you should ask yourself as you work towards your first presentation:

- **Can I summarize my goal in one sentence?** Try to make it clear enough that a colleague in the class would be able to carry out the same (or similar project) from this goal statement alone.
- **How will I know if I have been successful?** e.g. How will I measure the performance of my computer vision algorithm? Do I have a sharp hypothesis that can be tested experimentally, where the answer is going to be clear and simple?
- **What resources do I need, and will I be able to obtain all of them in the given amount of time?** Think about data (video, images), computational resources, human subjects, cameras etc.
- **Have I identified all the loose ends so that I can ask the TAs for help?** If there are loose ends be sure to highlight them during your proposal presentation.
- **Is my project designed in such a way that, no matter what happens, I will be able to report some results?** Is there the possibility that I will have nothing to report at the end? Describing a method but not having carried out the implementation / experiments / analysis is usually not sufficient to get a passing grade.

Collaboration

Teams of two students are ideal. You may also work by yourself. Teams of 3 students are discouraged. Teams of 4 or more are forbidden. The team works together in preparing the presentations, collecting and annotating data, writing the code, carrying out the experiments, and analyzing the data. <u>Each student is responsible for writing their own report</u>. Students from the same team may decide to analyze the data independently or carry out different experiments if they wish.

Checklist (dates may change):

DATE	WHAT
By April 25	Pick 1-2 interesting ideas for a project. Research them. Come up with a plan. Contact the relevant professor / TA and make sure that she/he is happy with the project.
April 28 and 29th, 7pm-8pm	**Proposal presentation due**.
May	Work on the project: collect data, analyze, model, implement….
June 3	**Final presentation due.**
June 3-15	Carry out last experiments following feedback. Write your final report.
June 15	**Final report due.**

Proposal presentation (2 slides)

Your presentation should include your plans to address the following questions:

1. What is the problem I am going to attack? What does the literature say? What are the open questions I am going to ask?
2. What is the plan? What data do I need to collect? Which algorithms am I going to implement and test? What is your minimal goal, what is your ambitious goal?

Make sure your slides are carefully crafted for effective communication. Here is a simple skeleton example that demonstrates what sort of information you might want to include.

Final presentation (5-8 slides)

1. Motivation, key questions, goal of project.
2. Previous work.
3. Technical approach, methods (1-3 slides).
4. Experiments (1-3 slides).
5. Conclusions.

Final report

Pick the format of a conference that seems appropriate for the subject matter (NeurIPS, CVPR, ICCV, ...). Pretend that you are publishing your results there (and if your project is interesting enough it will happen!). Use at most 8 pages (not including references).
Grading will be based on: (a) originality, (b) thoroughness, (c) clarity of exposition, (d) correctness, (e) completion (better finish a smaller project than stopping half-way on an ambitious project).
The writing should contain sufficient details that another student in the same class would be able to reproduce your results. No more than that. We do not need to see your code in the report. However, we encourage you to open-source your code on GitHub.

Figures you will likely want to include in the report:

1. Examples of images you are trying to analyze / classify
2. Diagram illustrating your approach
3. Error rate (training and validation) vs training epoch
4. Error rate vs n. Of training examples in log log scale
5. Examples of images that are annotated correctly
6. Examples of errors
7. Comparison of the performance of different variants of your method

Project ideas

Pick a topic that excites you. Make sure that the resources you need (e.g. well-annotated training and test data, computation) are easy to procure.

Just to get going, here are a few ideas. Feel free to come up with your own ideas. If you are not sure, we are happy to help you make your choice - the TAs are happy to help via email or office hours.

1. General Computer Vision
    a. Build a pedestrian detector.
        i. Caltech Pedestrians
        ii. Caltech Roadside Pedestrians
    b. Measure the performance of face detection, gender classification, face recognition etc. in cloud systems. Look for racial bias, gender bias, etc.
        i. NIST testing of face recognition systems: PDF - It's very well done but it does not work for cloud systems. If you are successful you would be the first independent entity to test commercial cloud systems (IBM, Google, AWS, …)
    c. Compute the pose (skeleton) of people who are working, playing, exercising, etc.
        i. MPII
        ii. Human3.6M

    d. Scene segmentation for autonomous driving.
        i. Cityscapes
    e. Classify art by style, period, etc.
        i. iMet Challenge
    f. Misc. Datasets
        i. COCO
2. Medical
    a. Image-based disease diagnosis / severity rating.
    b. Segmentation of tumors (in e.g. MRI) or cells (in pathology images).
    c. Assess social distancing compliance by counting the number of pedestrians in a video and estimate the distance between them.
    d. Misc. Datasets
        i. Diabetic Retinopathy Detection
        ii. Intracranial Hemorrhage Detection
        iii. COVID-19 Image Data Collection
3. Conservation and Animal Behavior
    a. Classify images of diseased plants.
        i. Plant Pathology Challenge
    b. Identify snake species to rapidly assess whether a snake is poisonous or not.
        i. SnakeCLEF Challenge
    c. Count the number of bees that are flying in/out of a beehive. Classify bees that are carrying pollen.
        i. Pietro’s Beehive Videos (upon request)
    d. Detect or classify animals in camera trap images.
        i. iWildCam 2020 Competition and Dataset
    e. Classify the species of animal in citizen science images.
        i. iNaturalist 2019 Competition and Dataset
    f. Predict land cover or species present from satellite imagery.
        i. GeoLifeCLEF 2020 Competition and Dataset