Computer Vision: Assignment 1

The field of Computer Vision has made important progress in many areas. However, it is far from being solved and many end-to-end systems require higher accuracy than current Computer Vision (CV) algorithms can provide. One field receiving increasing attention is to use human input (crowdsourcing) to provide the remaining level of accuracy.

Before you begin your research project, you should have an understanding of the broad areas that encompass the field of computer vision. What kinds of problems do CV researchers face? What tools are available to solve them? How can researchers evaluate their progress? How do they share their work?

In this first assignment, you will survey broad areas of computer vision to find specific applications where the facilitation of crowdsourcing can help make a difference.

Your task is to **write a short proposal** that outlines one potential area within the field of computer vision. Describe what is the problem that CV tries to solve and how crowdsourcing could help. Pick something that might be interesting to study for a few weeks.The field of Computer Vision has the tradition of hosting challenges to important topics. As a start, you should look for CV challenges as they typically represent important problems, are well defined and have existing solutions that can be build on.

Next week, you do **not** have to continue on this potential area that you propose. Feel free to switch to promising projects as you see fit.

Guidelines:

- This is essentially a sales pitch. Your goal is to sell us (and your fellow students) the idea of working on the problem you outline. Try to use concise, persuasive language to convince us that it's worth working on. The idea you propose here can be an idea everyone might work on, and you can play an important role in setting the direction of the project
- Start with the provided template as a guideline, and modify it to suit your needs. The example application should give you some idea of the level of detail we're looking for.
- Please don't go much longer than one page.
- **Note**: Include a short (~100 word) abstract summary at the beginning, so others can quickly skim your proposal to get a general idea of your project.

Here is a suggested template:

Describe the application. (The Summary of the Problem)

In the first paragraph, briefly describe the problem that you're trying to solve, drawing on one of the subfields of computer vision. After reading this paragraph, we should understand why this is a great problem that benefits the field.

- What real-world problem does this application try to solve?
- How does it help society?
- What obstacles are there to solving it? E.g. is the computation expensive? Are there not enough high-quality datasets? Do state-of-the-art approaches fail on simple cases?

Describe the process that is typically used to solve it. (The Design of a Typical System)

Many problems within computer vision are large and complicated, so researchers typically break them down into smaller problems that are easier to solve. A full system might look like a "pipeline" of algorithms and systems that solve small subproblems or tasks within the overall problem. The big problem is solved when all the tasks are completed. In this paragraph, outline the steps that a machine vision system typically takes to solve the problem you described. Briefly describe the algorithms that are used.

- How do existing approaches typically work?
- What is the 'pipeline' of steps?
- Is there a hardest step?

Describe how crowdsourcing could help. (The Solution)

This is the real meat of the proposal, since this is where you will spend most of your research times. Some of the pipeline steps you outlined above could be replaced by crowdsourcing, which would make the system perform better. Describe how your crowd workers could help the machine give better answers.

- What steps above could be replaced with crowdsourcing?
- What kinds of questions would you ask your crowd workers? What kinds of tasks would you have them perform?
- Why would this help performance?
- How much cost would this introduce?

Describe some previous work (The Literature Review)

In this section, give links to one or two state-of-the-art papers that help solve the problem. What tools do they use? Do they have a specific way of measuring their system's

performance? In many cases, you might be able to find a challenge dataset that several papers all report on.

This part doesn't need to be terribly detailed. In many cases, someone may have done most of the work for you by collecting all of the papers that report on a specific dataset, or by writing a survey paper that describes several common approaches in the field.

The point of this section is to give us and other students some jumping-off points and things to look for if we want to explore your problem more closely.

- What are some other papers that solve this problem?
- Are there any datasets?
- How is accuracy typically evaluated?

Example application: Face recognition

Face recognition is one of the best examples of computer vision. In this project, we propose using crowdsourcing to improve the accuracy of an automatic face recognition system, like the ones in Facebook. Humans can help computers recognize faces because humans are already born with an innate ability to do this, almost from birth, and not much human effort is required to get a reasonably accurate answer.

What's the problem? Recognizing faces

In this proposal, we tackle the problem of automatically recognizing a face in an image. This is a very common problem. For example, Facebook uses a face recognition system to help users "tag" images of their friends that they upload. Police can use face recognition to find suspects in surveillance videos.

Face recognition is currently very difficult for computers to do by themselves. There are several obstacles:

- Faces might be rotated in the image
- People can make strange expressions (sticking tongue out, etc) that confuse computer vision algorithms
- The image might be blurry

Describe the process that is typically used to solve it. (The Design of a Typical System)

Typically, a face identification process is broken down into several steps:

- First, the user must take a picture or upload an image
- Then, a "face detector" finds the location of all the faces in the image

- To recognize one of the detected faces, a "face pose estimation" step finds the location of the eyes, nose, and mouth of the face
- Typically, some algorithms try to "warp" the face into a reference image, eg. by rotating the face so that the eyes and mouth are in a common place
- A "feature extraction" step turns the face into a long sequence of numbers
- Then, a machine learning classifier takes the sequence of numbers as input and decide which person most closely matches the sequence.

Describe how crowdsourcing could help. (The Solution)

There are several ways that crowdsourcing could help here:

- Humans could help localize the face and find eye and mouth locations. If the pose estimation step fails, it's unlikely that the feature extraction or classification stages will work correctly because the image will be distorted. To avoid this problem, we could ask crowd workers to click on the eyes and nose of an uploaded face. This way, we can build actual face recognition systems that always can find the correct locations of the eyes and nose.
- Humans could help outline hard-to-see faces. In a blurry surveillance photo, some faces might be so hard to see that the first face detection step will never find them. In these cases, we could send some frames to the crowd and ask them if there are any additional faces that the detector could not detect.

Describe some previous work (The Literature Review)

Authors typically phrase the face recognition problem in two different ways:

- In the "Identification Setting", the input is a face and a list of people, and the system tries to guess which person the face belongs to.
- In the "Verification Setting," the input is a pair of face images. The system has a 'yes-or-no' choice: "Do these faces belong to the same person?"

One of the popular verification datasets is the "Labeled Faces in the Wild" (LFW) dataset, from the University of Massachusetts Amherst: http://vis-www.cs.umass.edu/lfw/index.html

Erik Learned-Miller lists several algorithms that report on LFW on the "LFW Results Page": http://vis-www.cs.umass.edu/lfw/results.html

Verification accuracy is typically reported as "TAR at FAR=0.01", which measures how often the algorithm says two identical faces are the same if the algorithm is tuned so that it only says two different faces are the same 1% of the time. Another way of visually showing accuracy is the "ROC curve," which graphs the true accept rate (TAR, how often the algorithm

says two identical faces are the same) as a function of the false accept rate (FAR, which measures how often the algorithm says two different faces are the same).

What should I write about? - to help you get started.

If you're having trouble thinking of an area to pursue, here are some resources you can look through:

- Listing of several startups in computer vision: http://www.cs.ubc.ca/~lowe/vision.html
- Wikipedia's list of computer vision applications and topics: http://en.wikipedia.org/wiki/Computer_vision#Applications_for_computer_vision
- Wikipedia's "Computer Vision Applications" category: http://en.wikipedia.org/wiki/Category:Applications_of_computer_vision

This list is not comprehensive.
Please note: You are welcome to propose anything that falls broadly within the domain of computer vision or digital imagery, as long as it's not closely related to our example proposal about "Face recognition". If you want to pursue something close to the example proposal, please explain how it is different.