Chapter 24

Lecture 24: Internet Vision

Weds, May 4, 2011 MIT EECS course 6.869, Bill Freeman

```
For the internet vision class, you can cover:
  Datasets and collection
    Mechanical Turk, CMU games, LabelMe
  Non-parametric methods
    Alyosha's filling-in paper
    80 million images
    infinite images
    video google (Josef's paper)
    future image synthesis and editing using large image collections.
  Approximate nearest neighbors
    LSH
    short codes

------------------------------------------------------

large datasets of labeled and unlabeled images.
 how to collect the large datasets

what you can do with them.
  alyosha, 80 million, infinite images

tools you need to work with them.
 approximate nearest neighbors
 short codes
```

How has the internet changed computer vision?

Many computer vision problems become very simple when large datasets are brought to bear on the problem. As Antonio mentioned, face recognition began to be solved when researchers applied large labeled datasets. The Microsoft Kinect uses a very simple feature set to identify body parts, and no spatial propagation of labels (no taking advantage of "shin-bone connected to the thigh-bone"... etc), but exploits a very large labeled training set of body poses and positions for training. This bears out the results of famous non-vision machine-learning based systems, such as Watson for Jeopardy and Deep Blue for chess.

We call this lecture "internet vision" because it addresses (a) how to collect very large datasets of images or videos, with full or partial labeling, and (b) what one can do with such datasets of images.

Two very large image datasets to mention at the outset is the photosharing web site, Flickr, and Facebook. For both datasets, many of the images are tagged, with keywords describing the photo, or with the names of the people in the photos. The Flickr tags are useful, but not always informative visually. Here's a Flickr group devoted to images of dogs named banjo. The keyword "banjo" won't be of much use to a visual object recognition algorithm here. You can imagine the rich dataset of all the tagged photos of friends and their friends on Facebook! People are just beginning to exploit this, see the paper by Stone et al.

24.1 collecting and labeling large datasets

Now let's turn to some internet-scale, labeled, or partially labeled, datasets that have been created.

24.1.1 ESP game

An early image labeling system that exploited the Internet is the ESP game of Von Anh and Dabbish. They created an appealing game, that people would be willing to play for free over long periods of time, that generated image labels as a by-product. You and your randomly assigned partner online, need to agree on words that describe an image. Often, there is a list of taboo words, which some other pair of players have already used to describe the image. People report that the game gives them a meaningful connection with their unseen partner. And image labels can be generated for countless images, essentially for free. Unfortunately, while there has been much enthusiasm for the prospect of millions of well-labeled images, I've never seen anyone use the label set, and as far as I know, the label set has not been made public.

24.1.2 LabelMe

At MIT, Antonio and his and my graduate student, Bryan Russell, created the popular crowd-sourced labeled image dataset, called Label Me. This dataset relies on people inputting a much richer labeling than with the ESP game: volunteers segment out objects of their choice. Some checking and cleaning occurs before objects are entered into the dataset, but the objects are named by the volunteers. This dataset is widely used.

24.1.3 80 million tiny images

Antonio and Rob Fergus, who was then my post-doc, made a leap forward in dataset size and comprehensiveness with the "80 million tiny images" project. There are several important elements to this project, which was a real struggle to get published, but now is recognized as an important contribution.

The first element is Antonio's observation of the ability people have to interpret even very tiny images of size 32x32. These images are "anti-SIFT" images–everything that rich feature descriptors like SIFT neet to interpret an image are not present in these tiny images. All that is present are global, contextual cues.

Slides 14, 15 shows an example of this. Subjects were presented with only the middle row of images–the 32x32 version of several scenes, shown to us (not to the subjects) at higher resolution in the top row. Asked to identify and label the objects in the image, humans produced the labelings shown in the bottom row. Note the pixel images of the things labeled! Clearly, the labelers are using their knowledge of how objects typically appear in scenes.

Slide 16. In fact, human performance in recognizing objects in 32x32 images beats computer performance at recognizing objects in 256x256 images. On the right hand side, we see ROC curves for the car recognition task from the 2006 PASCAL test dataset. Remember that up and to the left is better. The four circles show the results of human tests on the 32x32 sized images. The humans missed some of the very small cars, so performance wasn't perfect, but it was far superior to whatever machines could do then (and now).

The right hand plot shows the results for human subjects for scene recognition as a function of image resolution. Antonio will talk more about scene recognition next week, but the task of this test was to recognize what category of overall image was presented–street scene, cafe, beach, forest, etc. Using color data, people could do that task with absurdly low resolution images. There were 15 scenes to choose from, so chance performance was 7 percent.

Collecting internet scale image databases requires internet scale memory, which is prohibitive for an academic research group. But these tiny images, 32x32, offer a good compromise: there is enough in them to really work with them as images, yet they are small enough to let us store a vast library of images. Now the question is, how do we select and organize this large library?

Rob and Antonio elected to use WordNet as a framework for image collection. Wordnet offers a hierarchical structuring of English. As we'll show in the demo (follow the link in the slide), you can find definitions for every word, and a hierarchical expansion of each word–a car is a type of transportation device, which is a man-made object, which is an entity, for example (but see word net for the it's real expansion for "car").

They selected 70,000 non-abstract nouns, and queried a set of search engines for images for each of those nouns. This process took over 8 months, leading to about 1000 images for each word from this exhaustive set of images. Antonio made a great online demo letting users explore the set of images (follow the link).

So what can you do with an exhaustive collection of 32x32 images corresponding to nouns, each one noisily labeled with a query word? You can do lots of things, it turns out. For any given test image, it would seem natural to find a collection of the nearest neighbors of that test image. (There are choices in finding the nearest neighbor. One can get slightly better performance by shifting or warping the image to find the best pixel-wise match). It's interesting to see how that sibling set develops as we use a larger and larger subset of the 80 million images. Eventually, the set of images is large enough that we can find a collection of good matches to almost any image. (What do you think would happen if you tried to find a good match to an image of random noise?)

This raises the possibility–can you do object recognition simply by taking the nearest neighbors to your input image and reporting the objects in that nearest neighbor? Here's a study that tries to address that. In the test reported on the right, (c), human subjects were asked to report if two images corresponded to the same visual class. The probability of the answer "yes" is shown as a function of the pixelwise correlation between the two images. The 3 curves correspond to 3 different observers. So as we find images in the dataset with a pixelwise correlation of 0.9 or higher, there is a 50% or higher probability that the two images belong to the same object class.

Then we polled the dataset to see how the pixelwise correlation of the nearest neighbor image varies as a function of dataset size. You would expect that with a larger and larger dataset, the nearest neighbor images would have a higher and higher pixelwise correlation to the test image, simply because you'll have a better chance of finding a good match. That is the case, as shown in figures (a) and (b) of slide 23.

So let's now use this for object recognition, since it seems we may have a chance. To address the noisy labels of the 80 million images, we used a voting scheme: each of 80 nearest neighbors voted for each of its ancestors in the heirarchy tree. The results of that, for two test images, are shown in slides 24 and 25. Slides 26 and 27 give results for person recognition, given in 27 as a function of the fraction of the image that the person subtends. Recognition performance, especially for the small faces, is surprisingly good.

Slide 28 shows another application: automatic image colorization. The color component of nearest neighbors was applied to a high resolution input image.