

Image Search Engine

Sean Kelleher



Overview

Image search engine

Given a query image, return top-5 results

Raw pixels; no metadata

Unsupervised problem. But index + query labelled images = accuracy score



Dataset

Caltech-256

- 256 categories
- 25,000 train
- 6,000 test
- Reasonably challenging; high variation in objects, pose, lighting

Model

k-Nearest Neighbors

How to featurize images?

- Tried: raw pixels, Histogram of Gradients, and Convolutional Neural Net

What similarity (distance) metric for search?

- Tried: Euclidean and Cosine

Tools

Python: indexing, searching

Torch / Lua: convolutional neural net

GPU for the neural net

kNN was two lines of code

Bulk of the effort was writing tools!

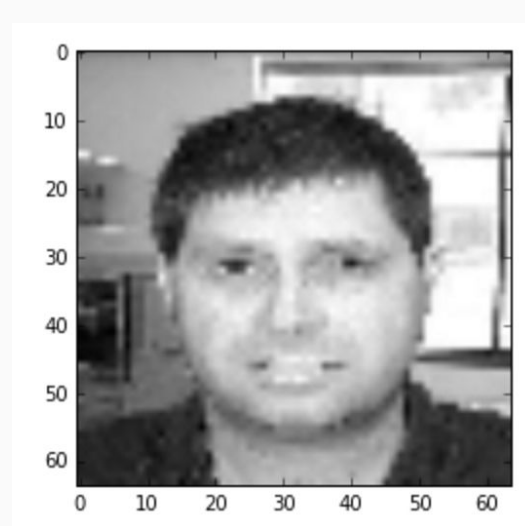
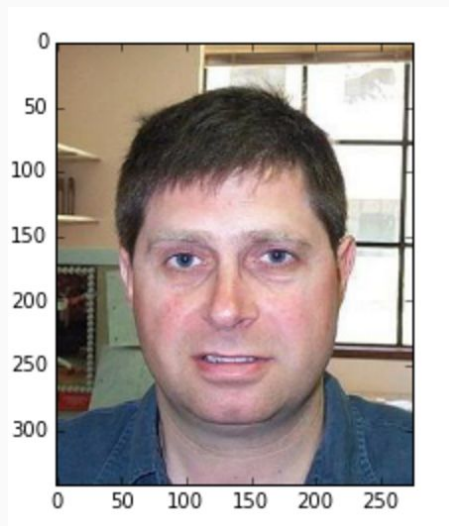
Wrote web servers to solve Python / Lua interaction.



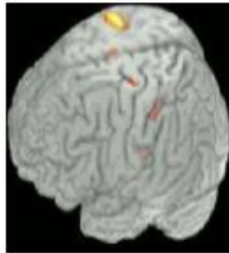
Features: pixels

Baseline:

- Resize to 64 x 64
- Grayscale
- Flatten to vector



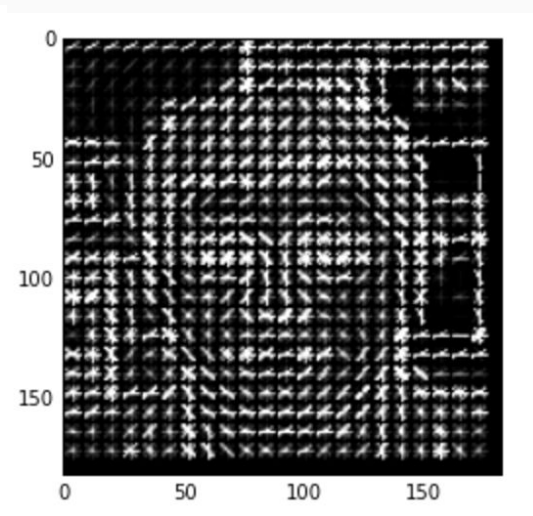
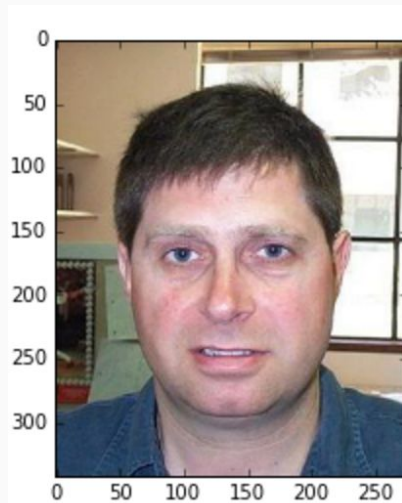
Features: raw pixels



Features: Histogram of Gradients

HoG:

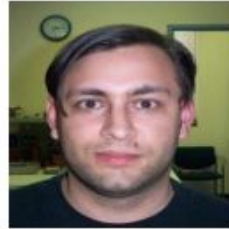
- Resize to 64 x 64
- Divide into cells
- Compute dominate gradients at each cell
- Compute histograms of gradients
- Flatten to vector



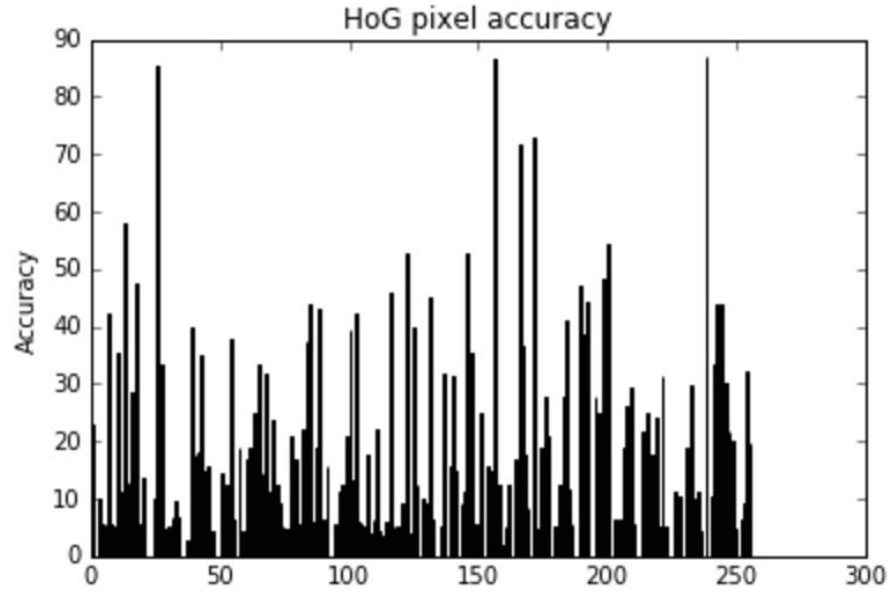
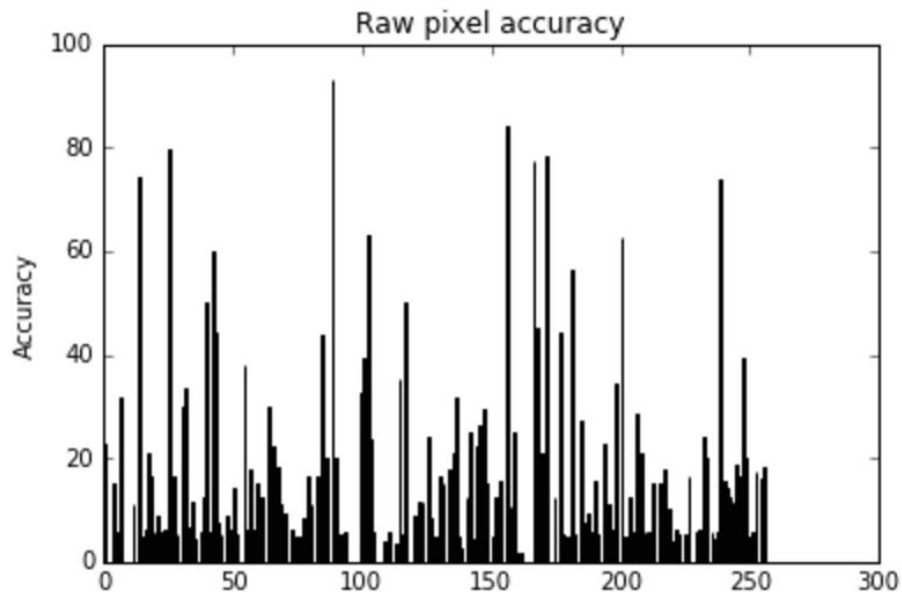
Features: Histogram of Gradients



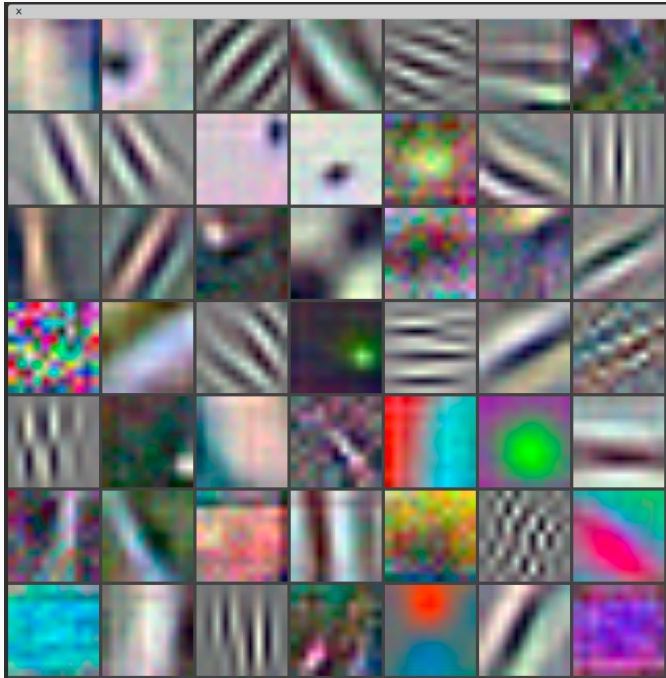
=



Features: Histogram of Gradients



Features: Convolutional Neural Net



Different Levels of Abstraction

Hierarchical Learning

- Natural progression from low level to high level structure as seen in natural complexity

Feature Representation



3rd layer
"Objects"



2nd layer
"Object parts"



1st layer
"Edges"



Pixels

Features: Convolutional Neural Net

Used pre-trained neural net that I had lying around

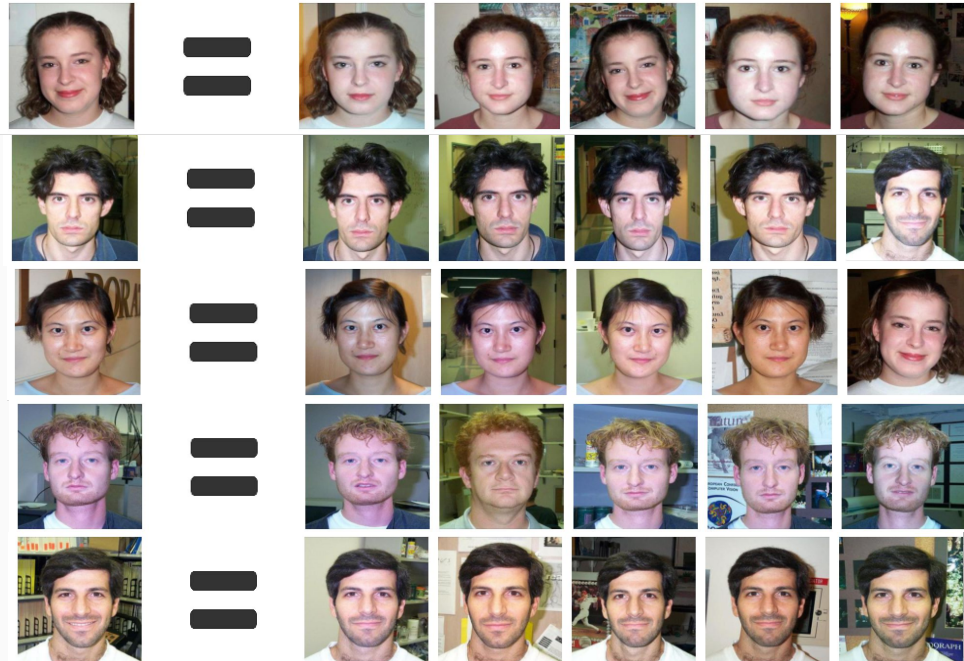
NOT trained on Caltech-256! Trained on other images!

Had learned generic image features

Simply chopped off the final layer (classifier)

Output was a vector of floats (the “deep features” of the image)

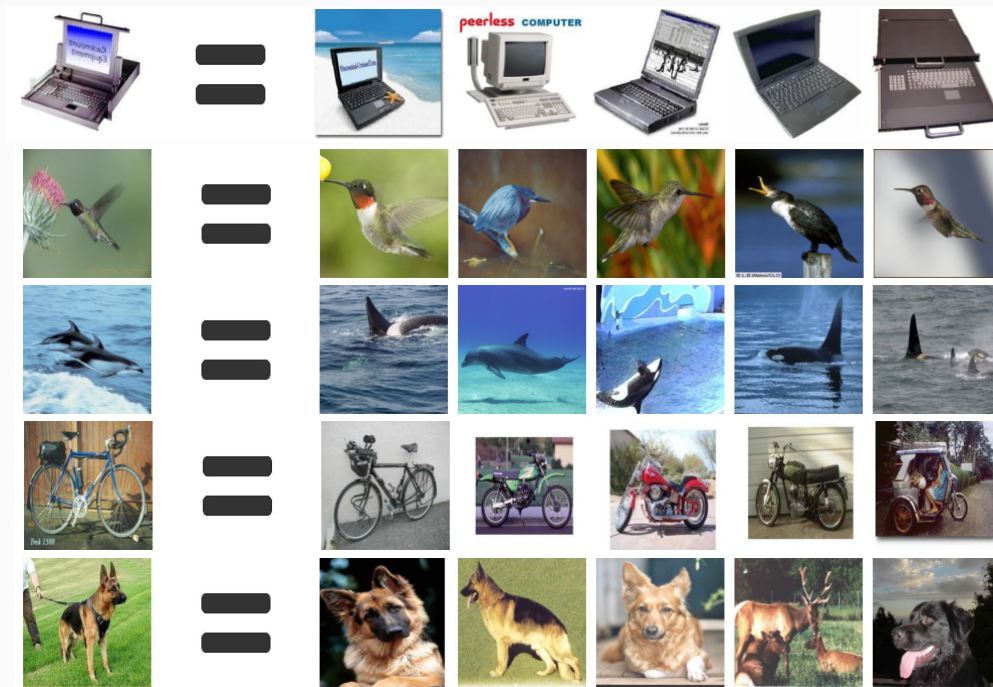
Features: Convolutional Neural Net



Features: Convolutional Neural Net



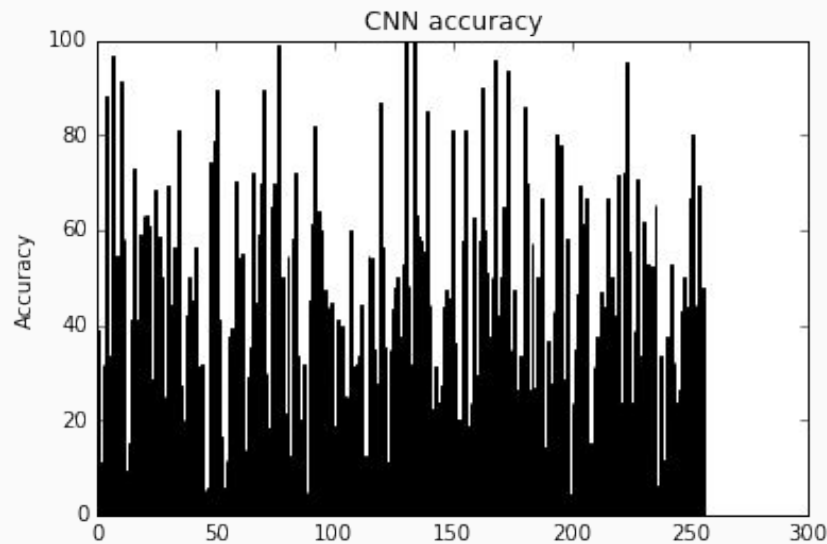
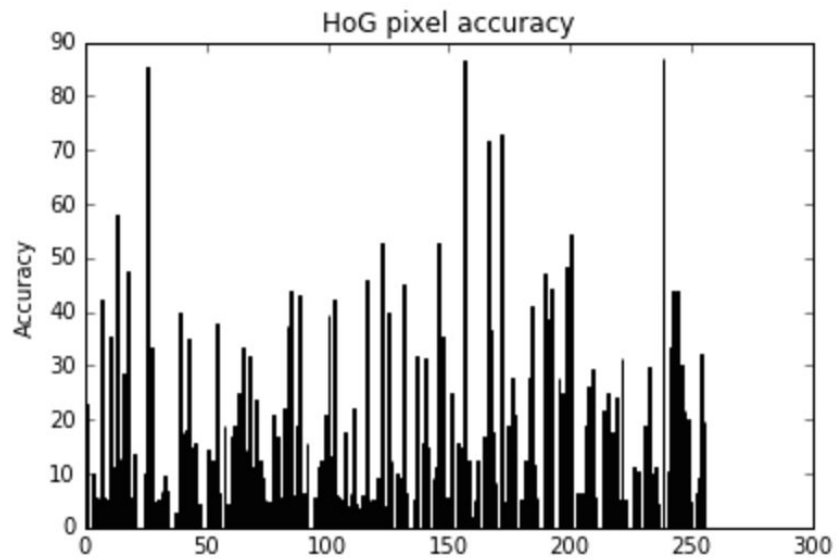
Features: Convolutional Neural Net



Features: Convolutional Neural Net



Features: Convolutional Neural Net



Features: Convolutional Neural Net

	Pixel/Euclidean	Pixel/Cosine	HoG/Euclidean	HoG/Cosine	CNN/Euclidean	CNN/Cosine
faces top-1	83.72%	84.88%	55.81%	82.56%	97.67%	93.33%
faces top-5	93.02%	89.53%	68.60%	91.86%	97.67%	93.33%
	Pixel/Euclidean	Pixel/Cosine	HoG/Euclidean	HoG/Cosine	CNN/Euclidean	CNN/Cosine
caltech256 top-1	11.33%	11.07%	11.48%	11.78%	27.25%	27.70%
caltech256 top-5	18.31%	17.92%	18.46%	19.56%	46.52%	47.90%

20% -> 48%

CNN dramatically improved top-5 search accuracy

(true accuracy arguably higher)

Thanks!

Sean Kelleher

sean@seankelleher.org

[@agent_cooper](#)

