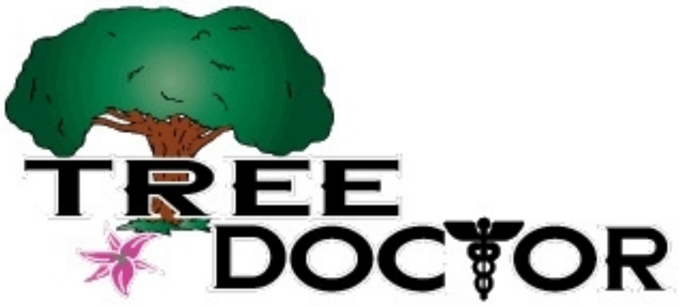
How would you approach the task to create a CV program that understands an aesthetics? For example, algorithm that picks up a well made professional logo out of several shitty ones

Libinpage **Apr 6**

Some times you don't need to be a professional designer to understand what was made by a good professional and where it's a amateur work.

Let's say you pick up a designer on freelancer.com or Fiverr. and you want to write a script that walk through a portfolio and pick up a professional designer. How would you approach this task?
here is two examples:



good
seattle
good business
network

Adrian Chief PyImageSearcher **Apr 6**

I have to say, this is a really, *really* hard problem mainly because it's so incredibly subjective. What may look shitty to you, may not look shitty to someone else -- aesthetics are a nightmare to deal with.

For something this complex, I think you would need *a lot* of data. Thousands, tens of thousands, potentially even hundreds of thousands of labeled good versus ugly logos. From there, train a CNN on the images and hope for the best. And even though, you'll really be at risk for overfitting. Computer vision is evolving very quickly, but I think we are still a long way away from having a good solution to this problem.

caioiglesias **Apr 26**

I saw a guy once picking nice girls on Tinder automatically based on their average face, which is "understanding aesthetics", but hardly applies to your logo case. I'll leave the link because I thought it was quite interesting.

http://crockpotveggies.com/2015/02/09/automating-tinder-with-eigenfaces.html

godreau **15d**

Seems like histogram would be a good choice here. Most modern logos are flat designed, meaning they only have a few colors in the palette. You can create a classifier based on those two logos above based solely on the number of distinct colors contained therein (for instance, the gradients within the crown of the tree). Not perfect, but may take your target data set from 1000 down to maybe 300.

¯_(ツ)_/¯