10-720 Computer Vision

Project Proposal

Title:

Personal Item Search from Single Image

Team Member: Yi Hua, Guanhang Wu

Abstract: Search personal item from single query. Comparing it with our database and outputs its attributes. For our project purpose, we start with searching watch because of its rigid surface and distinctive outlooks.

Approach: Our project consists of training phase and testing phase.
1. The training phase of our project can be divided into three parts: acquiring data, training a watch detector, and learning distinguishing features between different watch classes.

➢ Data
We will crawl Amazon or eBay for images of watch. Most of those images have a clean white background and shows enough details of the watch. Along with the images, we will also record the brand, price, and URL for the webpage.

➢ Watch Detector training
We plan on using HOG descriptor to distinguish a watch from non-watch photos. We will also experiment with SIFT, LBP, BRISK, FAST, and FREAK descriptors to see which would give the best result for watch detection.

➢ Class identifier
Ideally, we'd like to find the exact watch model given very few images of the watch. But if we cannot gather enough information to predict the exact model of the watch, we'd like to seperate watches into classes by their brand and price, so that we can find items similar to the query. We will combine distinctive features we found from the detector training phrase, and use SVM to classify training samples to differentiate class identifiers. R.Tao et. al. proposed to use data-driven non-semantic attributes for generic instance search.

2. On the testing phase, our project could be divided into two stages, generic search and discriminative search.

➢ Generic search
The first stage is generic search, which is to locate the watch in the image. We can divide the image into overlapping patches with multiple scales. Then, within each patch, we run our watch detector and do a non-maximum suppression to limit just one instance in the image. Meanwhile, we would set a threshold for the search results in order to eliminate the false positive samples.

➢ Discriminative search
The second stage of our testing phase is discriminative search, which is finding matches in our database. We would use class identifier to find its closest matches in our database. After finding its matches, we output their brand and price information for their reference along with the link to Amazon or eBay.

Dataset: Crawling from Amazon or eBay and store them in a database.

Software: C++ or Matlab

References

[1] R. Tao1, A.W.M. Smeulders, S. Chang, *Attributes and Categories for Generic Instance Search from One Example*, IEEE Conference on Computer Vision and Pattern Recognition.