Learning Discriminative Visual N-grams from Mid-level Image Features

Raj Kumar Gupta
gupta-rk@ihpc.a-star.edu.sg
Institute of High Performance Computing (A*STAR)
Singapore

Megha Pandey
pandeym@i2r.a-star.edu.sg
Institute of Infocomm Research (A*STAR)
Singapore

Alex YS Chia
alex.a.chia@rakuten.com
Rakuten Institute of Technology
Singapore

Mid-level image features have been shown to be helpful to bridge the semantic gap between low-level and high-level image representations. Many existing methods to learn mid-level visual elements consider each mid-level feature individually, and do not take their *mutual relationships* into account. We follow the intuitive idea that learning discriminative combinations of visual elements can help us deal with ambiguities better, and propose the concept of visual n-grams to effectively represent combinations of visual elements along with their relative spatial configuration and co-occurrence relationships.

An overview of our approach is shown in Figure 1. Figure 1 (a) shows the process of learning discriminative visual n-grams based on relative spatial position, orientation and co-occurrence relationships of mid-level image patches. Figure 1 (b) further shows how these visual n-grams are used to finally learn a feature vector representing test and training images.

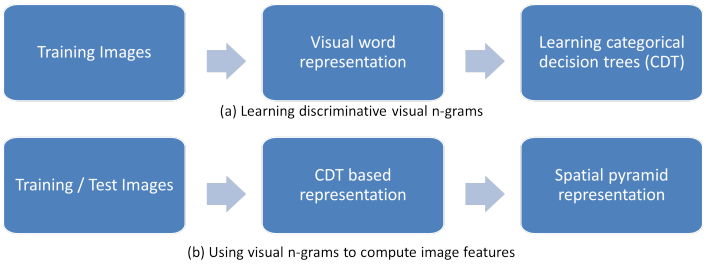

Figure 1: An overview of our approach. (a) Learning discriminative visual n-grams based on relative spatial position, orientation and co-occurrence relationships of mid level patches. (b) Using visual n-grams to compute feature representation for images.

We begin by densely extracting mid-size patches at different scales from the training images. Each patch is represented by a SIFT descriptor. We then learn a codebook by applying standard k-means algorithm. Each patch is then quantized to the nearest codeword representation in SIFT space. It is noteworthy, that while we use SIFT features for all the experiments in this work, our framework is generic and can be used with any other image descriptors as well.

The information about co-occurrence and relative positions and scales of different codewords is implicitly encoded by means of a *spatial co-occurrence vector*. For a given mid-level patch in the training set, we define a neighborhood over nearby grid locations and adjacent scales. The codeword indices of the patches in the neighborhood are concatenated to form the spatial co-occurrence vector for the given patch. Each dimension index of this vector refers to a particular position and scale relative to the current patch, while the value of the corresponding vector element captures the visual appearance of the respective neighboring patch.

Next, our goal is to learn combinations of mid-level elements that can best discriminate one image class from others. We employ categorical decision trees to represent and learn such combinations. Figure 2 shows a toy example for learning a categorical decision tree. For a given codeword c_0, we first locate its occurrences in the training images (Figure 2(a)), and extract spatial co-occurrence vectors for each of these locations. Figure 2(b) shows 4-dimensional vectors for illustrative purposes. In practice, these vectors are extracted over a larger neighborhood and multiple scales. Each of these vectors is weakly labeled with the label of the training image from which it was extracted.

Next, we learn a set of decision rules that can best separate this set of vectors into positive and negative instances, as illustrated in Figure 2(c). This set of decision rules can be encoded in the form of categorical decision tree, which is said to be anchored at the codeword c_0. *Each path from the root node to a leaf node of this tree is a visual n-gram*. In this manner, we can learn a categorical decision tree anchored at each of the codewords in

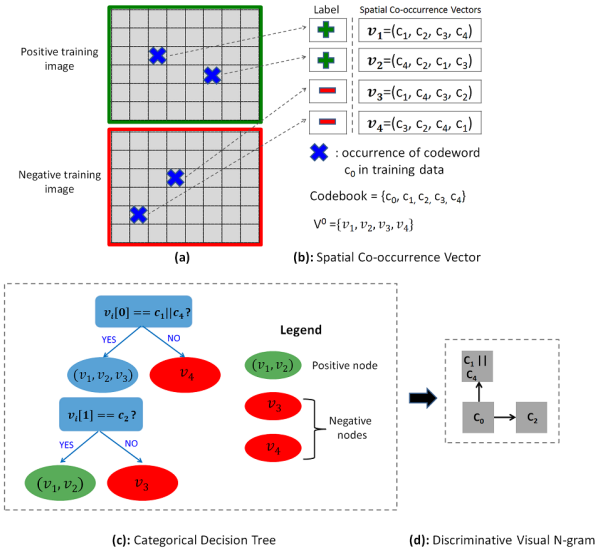

Figure 2: Illustrative example for learning a discriminative visual n-gram. Best viewed in color.

the codebook. We further employ multiple boosting iterations to ensure we effectively capture the diversity in the given set of images, and learn multiple categorical decision trees anchored at each codeword.

Having learnt a series of discriminative visual n-grams in this manner, we can now use them to compute feature representation for a given image. Given an image, we locate all patches represented by a particular codeword c_i, and extract the corresponding set of spatial co-occurrence vectors. These vectors are then classified using the categorical decision trees anchored at c_i, and the classification so obtained is used to compute feature values. Each visual n-gram i.e. each path from the root node to a leaf node in a tree contributes one feature value to the overall image representation. Once the feature values have been computed for all codeword occurrences in the image, we further use spatial pyramid representation to obtain the final image feature vector. The set of feature vectors so obtained is used in a standard SVM framework to perform image classification.

Dataset	Ours	Ours + IFV
Graz-01 (Average EER %)	94.0	95.5
INRIA horses (Classification rate %)	91.76 ±0.33	94.71 ±0.31
UIUC sports (Classification rate %)	83.54 ±0.41	93.12 ±0.28
Land-Use (Classification rate %)	79.52	87.24

Table 1: Classification results on four datasets using our image representation.

We have conducted experiments on four publicly available datasets: Graz-01, INRIA horse images, UIUC 8-sports events and Land-Use dataset. Table 1 lists the classification performance obtained using our feature representation on these datasets. We further combine our representation with global Improved Fisher Vector (IFV) features by concatenation, and include the resulting performance in Table 1 as well. Our method achieves high classification accuracy on each of these datasets. Our features also demonstrate excellent complementarity to global IFV features, in combination with which we outperform the state-of-the-art results on all four datasets.

To conclude, we have proposed an approach to learn discriminative combinations of mid-level elements by exploiting their spatial configuration and co-occurrences. Our method is, by nature, flexible to automatically learn a variety of combinations with different configurations and different number of visual elements. Our experiments demonstrate the effectiveness of the image representation so achieved.