Scene recognition via scene template learning

Haizi Yu
Computer Science Department
Stanford University
haiziyu@stanford.edu

Abstract

This paper concerns the problem of scene recognition in an image possibly containing a single or multiple scenes. The first half of the paper will concentrate on a newly built model, called scene template, and come up with a set of well developed methodologies to learn the model. The second half of the paper will introduce the technique used to recognize scenes, assuming an ideal model is given, or once the model is well trained.

1. Introduction

In daily life, people talk about objects individually, as well as thinking connections among objects. A practical way of treating different objects is performing a grouping task to the existing objects. This can be done in a hierarchical structure with individual objects lie in the bottom level, scene concepts in the top level, and the notion of object groups in between.

Both a bottom-up view and a top-down view occurs naturally and usually simultaneously when people are considering this hierarchical structure. On one hand, knowing the existence of certain objects helps us get aware of the scene content. For example, it is very likely that we are in an outdoor scene if we observe the existence of sky, trees and rivers; in contrast, it is very likely that we are in an indoor scene if we observe the existence of bed, desks and chairs. On the other hand, knowing the scene helps us both recognize and localize the individual objects better. For example, if we are told that we are in an indoor scene, it would be rare to expect trees in it. If we gain more knowledge about the scene, e.g., the prior information about the relative positions of objects in it, then it would be also rare to find there is a chair on the bed.

Object detection has been a hot topic in Computer Vision for a long time. However, through tons of research papers on object detection, people tend to treat the desired object individually and separately, as a single entity. In contrast, this paper will aim to detect a group of objects which tend to go together in our daily life, e.g., knife and fork, sky and cloud, in hope of exploring the relationship among objects. There comes the rough idea of object groups, also known as scene templates, as they are the building blocks of the concept of a scene.

Existing scene recognition techniques tend to treat an entire scene image as a training example and give it a hand coded label. The downside of this is the concept of a scene heavily relies on photographers personal interest, thus, it is very likely that the scene is not revealed in a natural way. One direct consequence of this is that it will be hard and confusing to give an image a label, if multiple scenes appear in the same image. So, this project tends to represent each image in the training set as a union of different scene templates. Together with the techniques developed in scene template learning, it will be possible in the end to extract any scene contained in an image and recognize it.

2. Statistics in a scene

This section will talk about common statistics (information) that contained in certain type of scenes. Two major issues will be talked about as listed below.

2.1. Connections between objects

Examples are plates and bowls tend to go together in a lot of images, however, it would be rare to see a motorbike and a bed go together in an image.

2.2. Relative locations between objects

In most situations, we would expect the desk lamp is on a desk instead of below it; the ceiling will appear in the upper half of an image while the floor will be in the lower half of an image.

3. Scene templates learning

An object group describes the concept of grouping a set of objects that share common features or exhibit a certain function as a whole. The task in this section deals with learning a set of object groups which forms a "basis" for the image world. This is analogous to finding a basis of a finite dimensional vector space in linear algebra. After learning, each image can then be represented as a vector based on the learned object groups, which is again analogous to computing coefficients of a vector in a vector space given the basis.

3.1. Notations

3.1.1 The big picture

Let $\mathcal{I} = \{I_1, I_2, \ldots, I_p\}$ denote the image world, where $I_i \in \mathcal{I}$ is a single specific image.

Let $\mathcal{G} = \{G_1, G_2, \ldots, G_g\}$ denote the set of g object groups, which is also known as a basis of the image world.

Our goal is then to learn $\mathcal{G}$ using $\mathcal{I}$.

3.1.2 Image and object group representation

Let $\mathcal{O} = \{O_1, O_2, \ldots, O_h\}$ denote a set of predefined individually existed objects represented by a string, e.g., $O_1 =$ "tree", $O_2 =$ "chair", etc.

Suppose $I_i \in \mathcal{I}$ is any image in the image world. We represent it as a three dimensional array, i.e., $I_i \in \mathcal{R}^{m \times m \times h}$. We use IMG_i with its natural pixel-valued representation to denote the same image as I_i.

We use the following two-step procedure to calculate I_i from IMG_i:

- Manually segment IMG_i into different parts, each of which denotes a certain object defined in $\mathcal{O}$. After this is done, the majority of pixels will be assigned an object label from an element in $\mathcal{O}$. For example, annotated images from LabelMe database are possible outputs of such procedure.

- Equally divide IMG_i both vertically and horizontally into $m \times m$ grids. For the image grid in the (x, y)th position ($1 \leq x, y \leq m$), we compute $I_i(x, y, z)$, where $1 \leq z \leq h$, as the percentage of pixels labelled O_z in that patch.

Suppose $G_j \in \mathcal{G}$. Then we represent $G_j \in \mathcal{R}^{n \times n \times h}$, where $n \leq m$ and $h = |\mathcal{O}|$ is the number of predefined objecs as mentioned above.

3.2. Problem formulation

At the beginning of this section, we made an analogy to vector space in linear algebra, where in order to get coefficients of an arbitrary vector given the basis, we project the vector onto each basis vectors using inner product.

Here, to compute coefficients of an image I_i given the basis $\mathcal{G}$, we use the maximum correlation of the image to each G_j's as opposed to doing projection using inner product in linear algebra.

More specifically, let $s_i \in \mathcal{R}^g$ denote the vector containing the coefficients that we want to compute. Then the jth coefficient corresponding to the jth basis object group G_j is calculated as follows:

$$s_i(j) = \max\{I_i * G_j\}, \quad j = 1, \ldots, g. \tag{1}$$

Then there comes the formulation:

$$\text{minimize} \quad \sum_{i=1}^{p} \frac{\|s_i\|_1}{\|s_i\|_2} - \frac{\|\bar{s}\|_1}{\|\bar{s}\|_2} \tag{2}$$

$$\text{subject to} \quad s_i(j) = \max\{I_i * G_j\}; \tag{3}$$

$$\bar{s} = \frac{1}{p}\sum_{i=1}^{p} s_i; \tag{4}$$

$$\sum_{k=1}^{h} G_j(:,:,k) = 1; \tag{5}$$

$$G_j(:,:,k) \geq 0. \tag{6}$$

3.3. Model explanation

3.3.1 G_j: ingredient analysis

3.3.2 $s_i(j)$: object group response

3.3.3 The S matrix

In this part, we are going to realign all the $s_i(j)'s$ into a big matrix, called S, and give a detailed analysis on the structure of S afterwards.

$$S \in \mathcal{R}^{p \times g}, \text{ where } [S]_{ij} = s_i(j). \tag{7}$$

A lot of researches have talked about the concept of feature distributions. Properties like **Population Sparsity**, **Lifetime Sparsity** and **High Dispersal** of feature distributions have been explored in the neuroscience literature. Methods designed based on these properties have been widely used and verified in machine learning, e.g., [papers on Sparse Filtering].

Our model heavily inherits the properties mentioned above to do a good job in the experiments. Put it in our language using the S matrix. We desire it to have the following properties.

- **Sparse features per image (Population Sparsity).**
- **Sparse features across images (Lifetime Sparsity).**
- **Uniform activity distribution (High Dispersal).**

It can be shown that Lifetime Sparsity can be implied once Population Sparsity and High Dispersal are pursued.

Then it is straight forward to encode Population Sparsity and High Dispersal in the objective (2), wherein the first term of the objective describes Population and the second term describes High Dispersal.

4. Recognition

In this part, we will show how to localize object groups in images and use the detected object groups for any scene recognition.

4.1. Automatic scene grouping

This section talks about automatically grouping images into scene clusters using the concept of scene templates.

4.2. Scene template localization

This section talks about experiments on detecting scene templates in images.

4.3. Scene retrieval

This section talks about methods used for scene retrieval, especially how the feature vector is designed in classification.