PLOS ONE

Differences Help Recognition: A Probabilistic Interpretation

Yue Deng, Yanyu Zhao, Yebin Liu, Qionghai Dai*

Department of Automation, Tsinghua National Laboratory of Information Science and Technology (TNList), Tsinghua University, Beijing, China

Abstract

This paper presents a computational model to address one prominent psychological behavior of human beings to recognize images. The basic pursuit of our method can be concluded as that differences among multiple images help visual recognition. Generally speaking, we propose a statistical framework to distinguish what kind of image features capture sufficient category information and what kind of image features are common ones shared in multiple classes. Mathematically, the whole formulation is subject to a generative probabilistic model. Meanwhile, a discriminative functionality is incorporated into the model to interpret the differences among all kinds of images. The whole Bayesian formulation is solved in an Expectation-Maximization paradigm. After finding those discriminative patterns among different images, we design an image categorization algorithm to interpret how these differences help visual recognition within the bag-of-feature framework. The proposed method is verified on a variety of image categorization tasks including outdoor scene images, indoor scene images as well as the airborne SAR images from different perspectives.

Citation: Deng Y, Zhao Y, Liu Y, Dai Q (2013) Differences Help Recognition: A Probabilistic Interpretation. PLoS ONE 8(5): e63385. doi:10.1371/journal.pone.0063385

Editor: Derek Abbott, University of Adelaide, Australia

Received January 24, 2013; **Accepted** April 2, 2013; **Published** May 31, 2013

Funding: This work was supported by the National Basic Research Project, No.2010CB731800, (http://www.most.gov.cn/tztg/200910/t20091023_73804.htm) and the Key Project of National Natural Science Foundation of China, Nos. 60932007 (http://nsfc.pubmed.cn/2ce238a81), U0935001 (http://nsfc.pubmed.cn/9f9338686), 61120106003 (http://nsfc.pubmed.cn/2966dd309), 61021063 (http://nsfc.pubmed.cn/f8d1a7607). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: qhdai@tsinghua.edu.cn

Introduction

Visual information understanding is a long-standing topic that has been extensively discussed in both the communities of vision research and artificial intelligence. Over last decades, a large number of prominent works have been devoted to this challenging task from diverse ranges of perspectives including biological investigation [1–3], psychological research [4–6] and computational methods [7–10]. In this paper, we propose a computational approach to reveal the differences of various images and will interpret how the differences help computers to conduct visual recognition tasks.

The fundamental observation that inspires our work is the behavior of human beings to distinguish different images. Psychologically, when distinguishing two images, we pay much attention to their differences rather than the common characters. For example, when categorizing two images from bedroom category and office category, the most representative information that helps recognition are the "beds" and "computers". When finding the patterns for beds, it is easy to tell that the image is from the bedroom. Similarly, when finding the features of a computer in an image, it probably describes an office scenario. However, in both of these two images, other common patterns are also collected, *e.g.* the wall and the ground. These common features help less to distinguish these two images because they appear in both of them.

Such desired information selection mechanism is easily implemented as visual information processing by human brain. Unfortunately, it can hardly be addressed by computers. Therefore, to fill the gap, we propose a probabilistic computational framework to model such behavior by statistical machine learning [11]. To better describe the computational model, we clarify two important concepts in statistics, *i.e.* generative model and discriminative model [12].

Loosely speaking, generative models, *e.g.* Gaussian Mixture Models (GMM) and the topic model, play the role of interpreting how the observations are generated/sampled from a probabilistic distribution. They are different from discriminative models in that generative models learn a joint probability distribution of both the observations and the corresponding labels. Generative models are used in machine learning for either modeling data directly (*i.e.* modeling observations drawn from a probability density function), or an intermediate step of forming a conditional probability density function.

In contrast, discriminative model generally reveals what makes data different among classes. In machine learning, discriminative model is always described by a conditional probability distribution which expresses the dependence of the labels on the observed features. Discriminative models differ from generative models in that they do not allow one to generate samples from the joint distribution of labels and features. However, for tasks such as classification, they do not require the joint distribution and, in most cases, can yield superior performance. Widely used discriminative models in machine learning include the Support Vector Machine (SVM), Multinomial Logistic Regression (MNL) [12] and Conditional Random Fields (CRF) [13].

In our work, a generative computational model is developed to describe how the image features in different classes are generated by a joint probability density function (pdf) of features and their labels. In addition, a discriminative model is incorporated into the Bayesian model to reveal the discriminative patterns among different classes. The whole model is solved in an Expectation-Maximization manner for parameter estimation and latent variables inference.

Using Bayesian model and probabilistic approaches for visual information understanding has been extensively discussed in communities of computer vision and machine learning. In [14], a Bayesian model is used for pattern detection, *e.g.* circle object, on images. However, their approach shed no light on the discriminative aspect. Probabilistic inspired metric [15,16] has also been incorporated into typical subspace models for visual feature extraction. But these approaches do not explicitly enhance the discriminative property of the extracted features. Different from existing works on image feature extraction, the model discussed in this paper is fully conducted in a Bayesian paradigm. Thanks to the flexibility of probabilistic models, the features selected from our model exactly reveal the discriminative information of an image in a much detailed manner.

In previous works, discriminative methods have been exploited into typical image coding framework to improve the image categorization performances. In [17] and [18], the discriminative functionality has been incorporated into the sparse coding method to learn the discriminative dictionary. However, the sparse coding methods pay too many computational costs to learning the dictionary and encoding one image, which requires solving the ℓ_1 type optimization for multiple times. In [19], a probabilistic based discriminative kernel has been defined to evaluate the distances of two images. It shed light on discriminative learning and could achieve very good performances by enhancing image differences. However, the work [19] only implicitly exploits the image differences and could not explicitly indicate what kind of features on the image are discriminative features. In our approach, thanks to the generative framework, we could exactly tell what kind of features cause the differences among images. Moreover, we will introduce how to construct the discriminative codebook by utilizing these image differences. Then, image encoding may follow the efficient paradigm of soft assignment [20].

To verify the effectiveness of the proposed framework, we apply the model on a number of image categorization tasks including natural scene recognition [21], earth observation (airborne images) recognition and indoor scene categorization [22]. Meanwhile, the experiment is conducted on bi-class classification and multi-class scene categorization tasks. It is glad to see from the bi-class categorization task that with the proposed framework, we can even accomplish the categorization work without training extra classifiers. The features learned by our model naturally exhibit significant discriminant structures for classification.

Materials and Methods

In this part, we describe our probabilistic model that interprets how the contents of different images are "generated". To describe the image contents, we follow the bag-of-feature method [23] to extract the image features at the pixel-level. In detail, a grid-based method is used to extract the dense SIFT features [24] from the images. The SIFT features are extracted on 16×16 pixel patches sampled every 8 pixels. For the ease of presentation, we define different image categories as $l=\{1...m\}$, which represents m classes. After extracting features from all the images, N feature-and-label pairs, *i.e.* (x_i,l_i) are obtained, where x_i records the SIFT feature and l_i denotes the label of a certain feature.

Method

In this paper, we consider local image features as two types: discriminative features and common features. The discriminative features are desired for classification purpose which significantly represent the class attributes. On the contrary, the common features appear in different categories and do not imply significant label information. To simultaneously model the generation of feature-and-label pairs, a probabilistic model is defined as

$$P(x_i,l_i|\phi_i)=[P_D(x_i,l_i)]^{\phi_i}[P_C(x_i,l_i)]^{1-\phi_i}, \quad (1)$$

where ϕ_i is the latent variable which equals to one if the i^{th} feature is a discriminative feature and zero if the feature belongs to the common feature set. In (1), $P_D(\cdot)$ and $P_C(\cdot)$ are used to represent the joint pdf of the feature-and-label pairs of the discriminative part and common part, respectively. Meanwhile, the probabilistic generative model for $P(x_i,l_i)$ is subject to the following expression,

$$P(x_i,l_i)=\sum_{\phi_i}P(\phi_i)P(x_i,l_i|\phi_i)=P(\phi_i=1)P_D(x_i,l_i)+P(\phi_i=0)P_C(x_i,l_i). \quad (2)$$

To mathematically describe the joint pdf of discriminative features and their labels, *i.e.* $P_D(x_i,l_i)=P_D(x_i)P_D(l_i=k|x_i)$, the conditional probability $P_D(l_i=k|x_i)$ should be explicitly defined. These discriminative features are class-specific and are different between categories. Therefore, this kind of features imply prominent label information and, mathematically, can be well explained by a discriminative functionality. In this paper, we exploit the Multinomial Logistic Regression (MNL) to model the conditional probability $P(l_i=k|x_i)$ due to the following two reasons.

First, an MNL explicitly minimizes the logistic empirical losses as the objective whose performances have been widely admitted in a number of practical applications [25,26]. As indicated in [12], the accuracy of MNL is similar to support vector machine (SVM) in addressing many real-world tasks. Meanwhile, the remarkable advantage of MNL for our case is its explicit probabilistic definition and its ease of being incorporated into our probabilistic model seamlessly. Accordingly, we define the pdf for the discriminative part as,

$$P_D(x_i,l_i|\theta)=P_D(x_i)\frac{\exp(w^{(k)}x_i+b_k)}{\sum_{j=1}^{m}\exp(w^{(j)}x_i+b_j)}, \quad (3)$$

where $\theta=(w,b)$ is the parameter of the MNL. Besides, a Gaussian mixture model (GMM) is adopted to encode the prior of $P_D(x_i)=\sum_{t=1}^{m}\pi_t\mathcal{N}(x_i|\mu_t,\sigma_t)$, where π_t is the mixture coefficient and $\mathcal{N}(\cdot|\mu,\sigma)$ is the Gaussian distribution with μ and σ as the mean and the standard deviation, respectively. In this paper, we select the number of mixture components as m which corresponds to the number of classes.

Similarly, it is easy to model the common feature generation part as,

$$P_C(x_i,l_i)=P_C(x_i)P_C(l_i|x_i). \quad (4)$$

In the above equation a GMM model is used to model the prior of $P_C(x_i)=\sum_{h=1}^{c}\pi_h\mathcal{N}(x_i|\hat{\mu}_h,\hat{\sigma}_h)$, where c is the number of

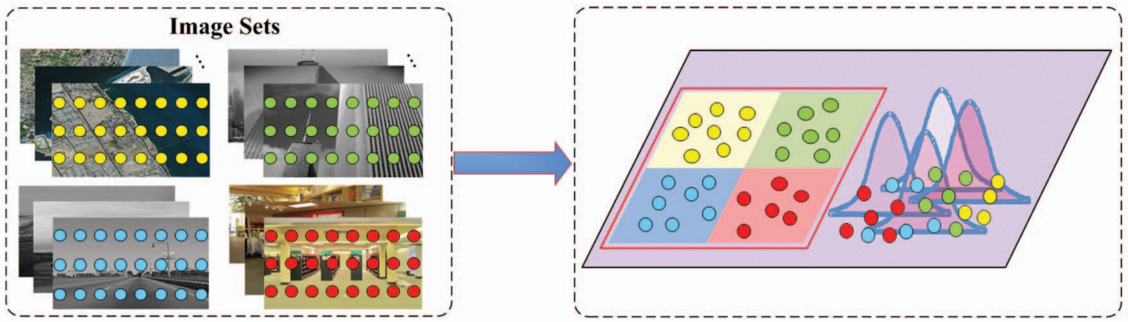


Figure 1. An overview of the proposed algorithm. In the figure, the round dots represent the extracted image features (local SIFT features) and different colors represent different category attributes of the image features. In the left panel, the local image features are extracted from all the images in multiple classes at the pixel-level. In the right panel, the proposed algorithm automatically determines what kind of local features represent the differences between classes and what kind of features are common ones in multiple classes. The features in the red rectangular are the discriminative features which are well modeled by a Multinomial Logistic Regression model. Besides, the common features in multiple classes do not take significant class information and are modeled by a Gaussian Mixture model.
doi:10.1371/journal.pone.0063385.g001

components selected in the model. Besides, it is interesting to note that (4) describes the common features in multiple classes, therefore, the feature does not exhibit significant category attribute. Mathematically, the feature is independent to the label and thus $P_C(l_i|x_i)=P(l_i)=\frac{1}{m}$, where m is the number of classes. Till now, we have explained both the discriminative part and the common part of the model. An overview of the proposed method is provided in Fig. 1.

Here, we remark on the parameter set Γ used in the model. $\Gamma=\{\theta,\mu,\sigma,\hat{\mu},\hat{\sigma}\}$, where $\theta,(\mu,\sigma)$ are the parameters for the logistic regression and GMM in (3), respectively. $(\hat{\mu},\hat{\sigma})$ is the parameter of the GMM in (4). In statistics, to estimate the parameters, a maximum likelihood (ML) formulation should be maximized,

$$\Gamma=\arg\max_{\Gamma}\{\log P(\mathcal{X},\mathcal{L}|\Gamma)\},\tag{5}$$

where $P(\mathcal{X},\mathcal{L}|\mathit{\Gamma})=\prod_{i=1}^{N}P(x_i,y_i|\Gamma)$. Unfortunately, the above log-likelihood is not solved analytically due to the incorporation of the latent variable $\Phi=\{\phi_1...\phi_N\}$. In this paper, we use the Expectation-Maximization (EM) algorithm to efficiently solve the model.

Table 1. The EM algorithm for model inference and learning.

```
Input: Feature points $X = \{x_1,...x_N\}$ and their labels $L = \{l_1...l_N\}$
Initialization: $k:=1,\Gamma^1:=\Gamma^{\text{ini}},u_i^1=\bar{u}_i^1=\frac{1}{2}$.
Repeat
  E-Step:
    Estimate $u_i^k$ and $\bar{u}_i^k$ according to (7);
    Calculate $Q(\Gamma,\Gamma^k)$ according to (6) with the current estimation of $u_i^k,\bar{u}_i^k$;
  M-Step:
    $\Gamma^{k+1}=\arg\max_{\Gamma} Q(\Gamma,\Gamma^k)$;
    $k=k+1$;
Output: $u_i^k$.
```

doi:10.1371/journal.pone.0063385.t001

An EM solution

EM method is widely used in statistics to solve probabilistic models with latent variables [27,28]. In [29], it is proven that EM method establishes the lower bound of the likelihood in (5) using the Jensen's inequality. Then, the lower bound is iteratively maximized and the whole optimization is solved by alternating between two steps, *i.e.* E-step and the M-step.

- In the E-step, the conditional expectation for the complete data likelihood is determined with the current estimation of parameters.
- In the M-step, the parameter is updated by maximizing the conditional expectation in the E-step.

The EM algorithm is a special case of the more general Majorization Minimization (MM) algorithm [30]. In our model, we first calculate the conditional expectation for the complete-data likelihood [31],

$$\begin{aligned}Q(\Gamma,\Gamma^k)&=E_{\Phi|\mathcal{D},\Gamma^k}\{\log P(\mathcal{D},\Phi|\Gamma)\}\\&=E_{\Phi|\mathcal{D},\Gamma^k}\left\{\sum_{i=1}^{N}\log P(\mathcal{D}_i,\vartheta_i|\Gamma)\right\}\end{aligned}\tag{6}$$

Here, for the ease of illustration, we define the data set as $\mathcal{D}=\{\mathcal{X},\mathcal{L}\}$. Then, in the M-step, the Q-function is maximized to get the new parameter, *i.e.* $\Gamma^{k+1}=\arg\max_{\Gamma} Q(\Gamma,\Gamma^k)$. The E-step and the M-step are iteratively processed until the convergence is reached. In our simulation, the convergence criteria is regarded as satisfied when the change of the objective between two iterations is below $1e^{-3}$.

In our model, one critical step is to estimate conditional probability for the latent variables given the model parameters. We define u_i and $\bar{u}_i$ as follows,

$$u_i=P(\phi_i=1|x_i,l_i)=\frac{P(x_i,l_i|\phi_i=1)P(\phi_i=1)}{P(x_i,l_i)}\text{and}\bar{u}_i=1-u_i.\tag{7}$$

Physically, u_i represents the probability that the i^{th} feature is a discriminative feature and, on the contrary, $\bar{u}_i$ records the

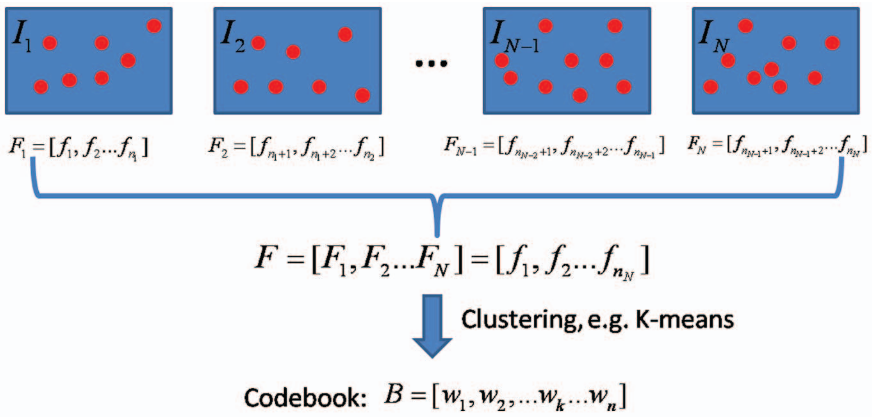

I_1
I_2
I_{N-1}
I_N
$F_1=[f_1,f_2\dots f_{n_1}]$
$F_2=[f_{n_1+1},f_{n_1+2}\cdots f_{n_2}]$
$F_{N-1}=[f_{n_{N-2}+1},f_{n_{N-2}+2}\cdots f_{n_{N-1}}]$
$F_N=[f_{n_{N-1}+1},f_{n_{N-1}+2}\cdots f_{n_N}]$
$F=[F_1,F_2\dots F_N]=[f_1,f_2\dots f_{n_N}]$
Clustering, e.g. K-means
Codebook: $B=[w_1,w_2,\dots w_k\dots w_n]$

Figure 2. The generation of a codebook with n codewords. The blue rectangle means the training images and the red dots represent the local features on the image.
doi:10.1371/journal.pone.0063385.g002

probability that the i^{th} feature belongs to the common feature set. Accordingly, we present the whole EM framework in Table 1. The detailed derivations for the EM algorithm and parameter updating rules are provided in File S1.

To solve the EM model in Table 1, the MNL in the discriminative part is initialized by fitting the original data with a logistics regression. In the other part of the model describing the common features, a Gaussian Mixture Model is applied to fit all the local image features, and then the obtained parameters are used for initialization.

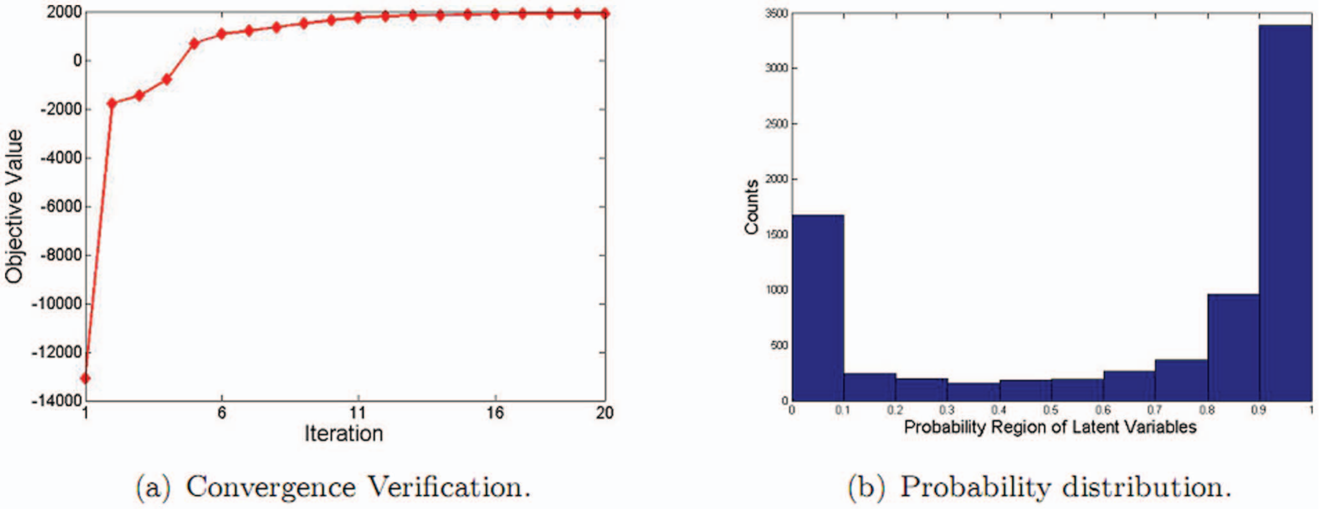
Feature Encoding by Exploiting Image Differences

In previous parts, we have stated how to determine the discriminative patterns from multiple images. However, the final goal for many scene recognition tasks is for categorization. The selected features are only local features (extracted at the pixel-level). Therefore, to capture the information of the whole image, some further coding steps are desired to generate features at the image level. In computer vision, one prevalent method for image coding is the bag-of-feature method in which all the local features on the images are assigned to a dictionary with multiple codewords. To make this paper self-contained, we briefly review the procedures of bag-of-feature method for image-level feature encoding.

As discussed previously, on one image, we can generate many local SIFT features. These local features cannot be directly used for image-level understanding. To describe an image, we should generate an image-level feature by exploiting the local descriptors on it. One prevalent approach to construct such an image-level feature is to construct a codebook and then assign local descriptors to the codebook, forming the image-level feature for classification.

Codebook construction is a training procedure, which clusters bags of local features, *e.g.* SIFT features, on the training images into n representative codewords. Then, during the second step, the features on the test images are assigned to their corresponding codewords in the codebook. An image-level feature, *i.e.* a histogram, is accordingly generated during codewords assignment to represent an image.

Fig. 2 summarizes the main procedures of codebook generation. Firstly, the local features in each image should be extracted and then thousands of local descriptors from multiple images are clustered into n centers. These centers are the codewords in the



(a) Convergence Verification.

(b) Probability distribution.

Figure 3. Numerical Verifications. (a) the objective value with the iterations and (b) the probability distribution of discriminative significance indicator u.
doi:10.1371/journal.pone.0063385.g003

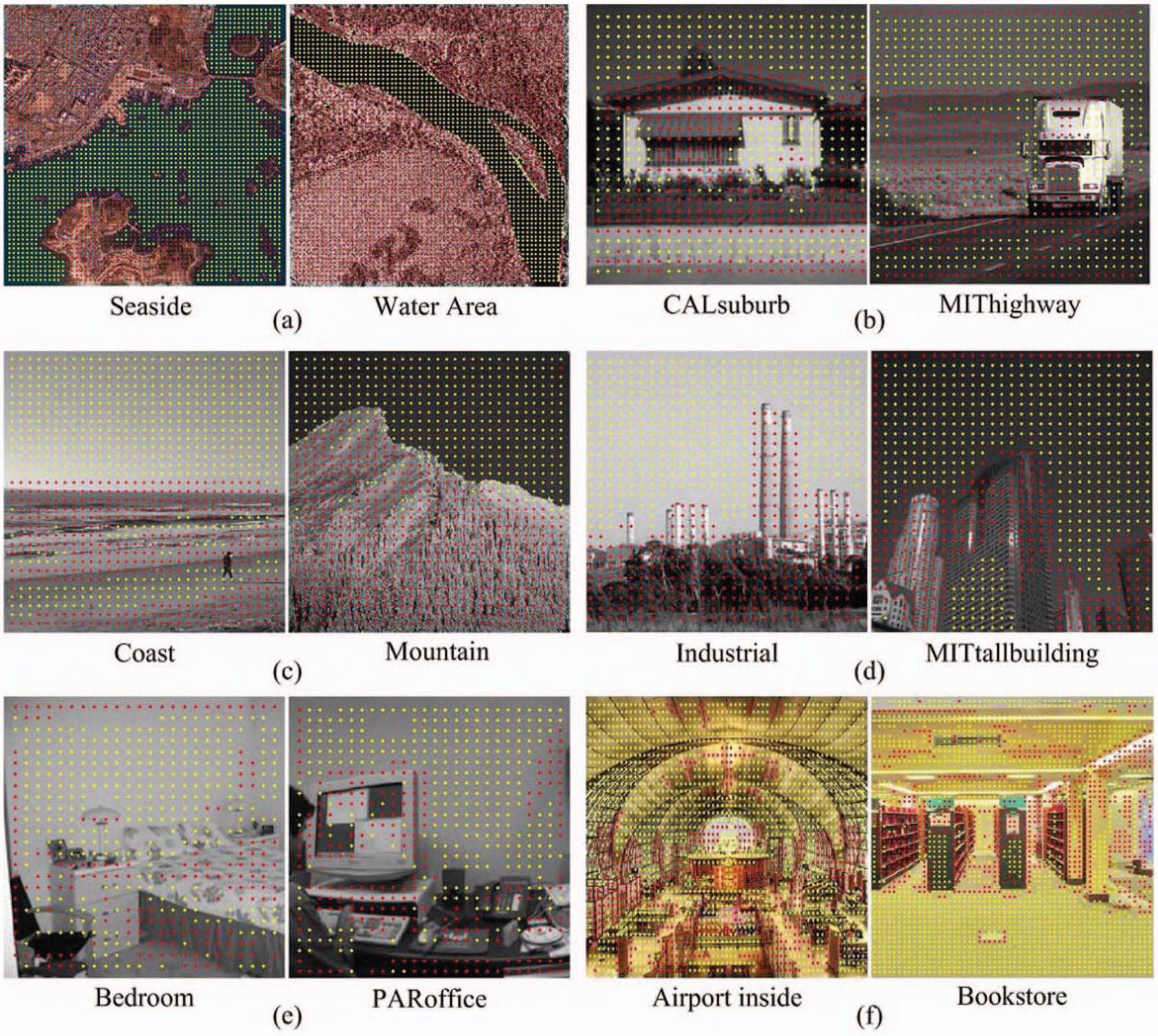


Figure 4. Sample images and discriminative feature selection results on bi-class categorization tasks. In each sub-figure, the red dots represent differences of the two classes and the yellow dots are their common contents.
doi:10.1371/journal.pone.0063385.g004

codebook. Therefore, the codewords in the dictionary are not determined by any specific local feature. Instead, they capture the common properties of all the local features from training images. After constructing the codebook, the local image features are assigned to the codewords to form an image-level histogram. In this paper, we adopt the soft assignment method introduced in [20] to conduct codeword assignment.

From the above discussions, it is apparent that the codewords play very critical role in generating an image-level feature. Therefore, it is desired that the generated codewords could enhance the discriminative information of different images. Fortunately, the proposed Bayesian model has provided feature types (*i.e.* discriminative or common features) for all the local features. Accordingly, it is nontrivial to exploit such prior to generate more descriptive codewords for image categorization. In typical codewords generation, all the local features from the training images are used to generate the codewords by clustering. In this work, the codewords are only clustered by those discriminative local features selected by our algorithm. We regard x_{i} as a discriminative feature if $u_{\mathrm{i}} > \tau$ and τ is fixed as 0.9 in our simulation. The selected discriminative features reveal the differences among multiple images. Therefore, when using these discriminative local features for clustering, the centers, *i.e.* the codewords, may better reveal the discriminative information on the images.

Results

In this part, we experimentally verify the performance of the proposed statistical model on treating real world images. We will report some numerical properties of the statistical model and then two experiments on bi-class and multi-class classification will be respectively conducted to verify the effectiveness of the selected discriminative features for image categorization. But before the experimental discussions, we first state the datasets that are used in the experiments.

Data Collection

The data are from three datasets, Synthetic Aperture Radar (SAR) dataset, Fifteen Scene dataset and MIT Indoor Scene dataset, among which SAR dataset is established by us.

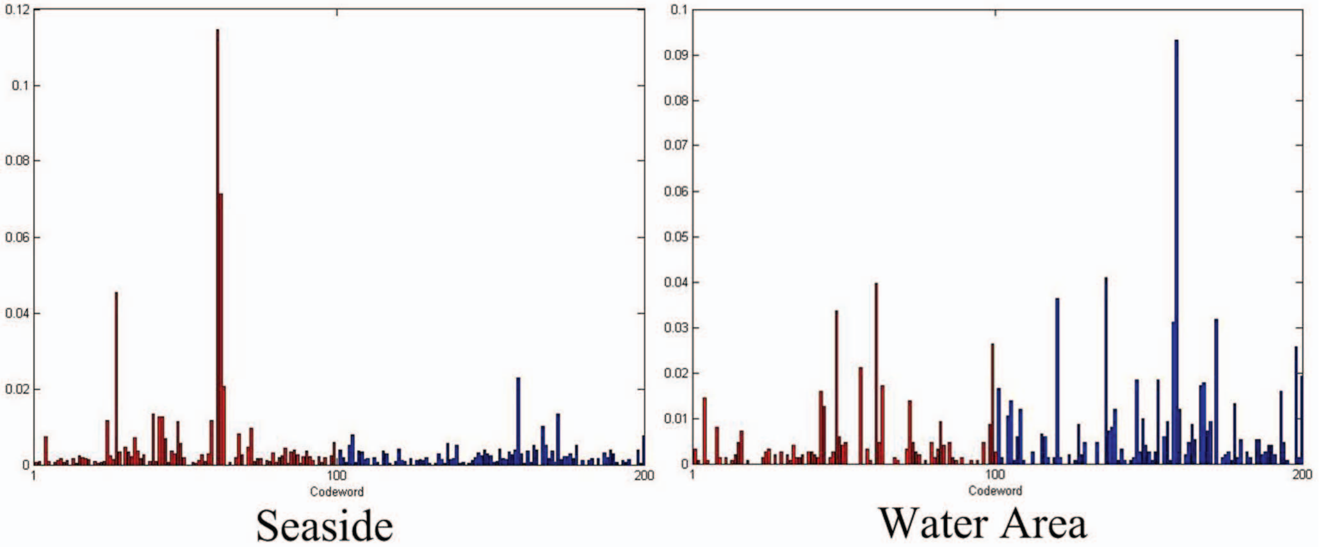


Figure 5. Histograms of two images in Fig.4 (a) generated using discriminative codewords. The red region represents the codewords from the seaside category and in the blue region are codewords from water area category.
doi:10.1371/journal.pone.0063385.g005

SAR consists of six ground categories: city, countryside, mountain, river, seaside, and water area. Each category in SAR contains more than 50 images. Most of these scene images are captured in China with our own airborne SAR devices. The original SAR data are converted into gray scale images and the experiments are only performed on these pseudo-gray-scale images. These images are cropped from a large SAR map to around 500×500 pixels.

The Fifteen Scene is a dataset of fifteen natural scene categories released by Fei-fei *et al* [21]. The image classes include bedroom, suburb scene, industrial, kitchen, living room, coast, forest, highway, inside city, mountain, open country, street, office, store, and tall building. Each category in Fifteen Scene dataset contains over 200 images. This dataset has been widely used in the field of computer vision as a benchmark dataset for scene categorization. In our experiment of bi-class recognition, most of the image classes are from the Fifteen Scene dataset.

In high level vision, indoor scene recognition is a challenging open problem. Most scene recognition models that perform well with respect to outdoor scenes work poorly, however, on indoor scene recognition. The MIT Indoor Scene dataset is released by Quattoni *et al.* in [22]. The dataset contains 67 indoor categories, and a total of 15620 images. The dataset contains indoor scene images ranging from airport inside, bookstore, library, mall, and video store to other scenes such as warehouse, dining room, fast food restaurant, and computer room. The number of images varies across categories, but there are at least 100 images for each category. This dataset has become a benchmark for indoor scene recognition.

Numerical Performance

For numerical analysis, we first report some results associated with the convergence of the learning model. As our statistical model is solved with an EM framework, we first show the objective value of the Q-function, *i.e.* Eq. (6), in Fig. 3 (a). From the result, it is obvious that the Q-objective function of EM algorithm increases along with the iterations. It is also apparent that the whole optimization framework reaches convergence within 20 iterations.

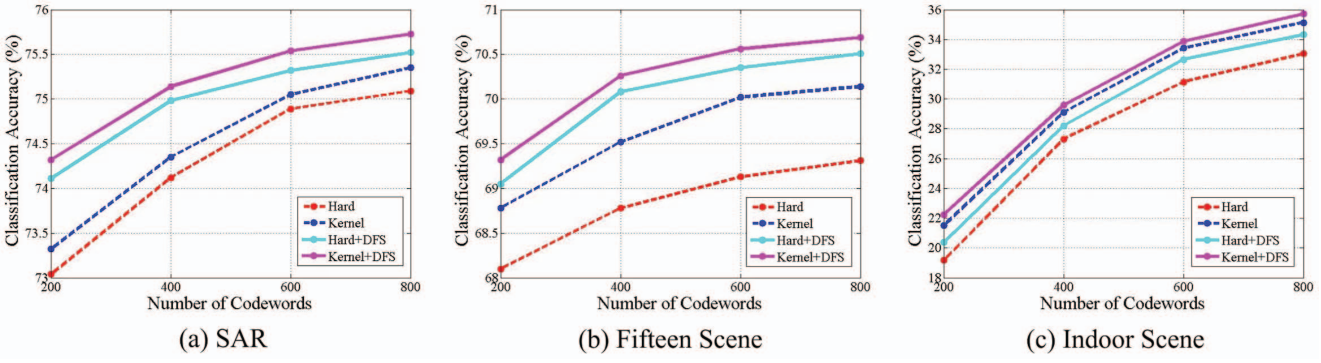
Besides, we also report the learning results of the whole EM algorithm. The basic pursuit of Table 1 is to find the features that capture significant label information of different classes. With this purpose, latent variables u_i are introduced for each feature in the training process. According to Table 1, the discriminative saliency variables u_i are the desired output of the probability model. If u_i is large, the corresponding feature x_i is probably a discriminative feature whereas a small u_i implies that x_i is a common feature shared by multiple classes.

To find the numerical property of the learning results, we conduct experiments on a bi-classes classification task. We apply our model to the Seaside and Water Area classification task of Fig. 4 (a). The statistical results of the distribution of u_i are reported in Fig. 3 (b) where the abscissa divides the probability of [0,1] into ten regions. The ordinate records how many local features in the training set fall into the corresponding region. From

Table 2. Recognition Accuracy of Bi-class Classification Tasks(%).

Classes	Seaside/ WaterArea	Bedroom/ PARoffice	CALsuburb/ MIThighway	Industrial/ MITtallbuilding	MITcoast/ MITmountain	Airport inside/ Bookstore
DC+FSC	90.3±0.7	83.4±0.8	**94.2**±0.7	87.8±0.6	88.1±0.8	73.7±0.6
DC+SVM	**91.8±0.6**	**89.5±0.5**	**93.4±0.6**	**88.1±0.7**	**91.5±0.5**	**76.4±0.6**
TC+SVM	88.41±0.5	85.1±0.7	90.3±0.5	85.5±0.7	89.3±0.7	72.9±0.7

doi:10.1371/journal.pone.0063385.t002



(a) SAR (b) Fifteen Scene (c) Indoor Scene

Figure 6. Classification accuracy on three datasets with different numbers of codeword.
doi:10.1371/journal.pone.0063385.g006

the results, it is obvious that most features are separated into two groups of discriminative features (with large u_i) and common features (with small u_i), respectively. The latent variables larger than 0.9 indicate that their corresponding features are very likely to be discriminative ones and will probably help recognition. In contrast, those on the left with probability lower than 0.1 denote common features in both classes, whereas these features have little contribution to the recognition. The results also serve to verify that our statistical model could exactly distinguish these two kinds of features from practical images.

Results on Bi-class Recognition

In this subsection, the proposed model is implemented on real image classification tasks. First, we consider performing the discriminative feature selection experiments on bi-class recognition cases.

Six pairs of classes that cover indoor, outdoor and aerial scenarios are tested in the experiment. For each pair of classes, our model tries to select out the common features as well as the features that are more distinguishable for classification (discriminative features). The sample images of each classification task and our feature selection results are provided in the sub-figures (a) to (f) in Fig. 4. In these images, the red dots represent the discriminative features (*i.e.* the differences) and the yellow dots are common features shared by two classes. Specifically, in our experiment, all features from images of each class are included and clustered into 4000 centers by K-means. Then, the clustered features (as many as 8000 for two classes) are included in the training process of the model. Discriminative and common clustered features will be accordingly determined by the estimation of their latent variables after the convergence of the EM algorithm.

We analyze the visual results in Fig. 4. In sub-figure (a), the images of Seaside class and Water Area class both have water regions. Therefore, the water regions are selected as common features and other contents are labeled as discriminative features. This observation is also found in Fig. 4 (b) to (d) that the sky in these images are selected as common features and other contents carrying significant category information are denoted as differences. In Fig. 4 (e), for the indoor scenario, the walls are regarded as common features while the bed in bedroom class and the computer in office class are selected as discriminative features. Finally, when distinguishing the images from airport class and bookstore class in Fig. 4 (f), the ground and the wall provide less category information. The discriminative patterns mainly distribute on the counter (in airport scenario) and books (in bookstore scenario).

Table 3. Recognition Accuracy of Multi-class Classification Tasks Using Spatial-Pyramid(%).

Datasets	Hard	Kernel	Hard + DFS	Kernel + DFS
SAR	81.2±0.9	82.9±0.7	84.4±0.6	87.3±0.5
Fifteen Scene	75.8±0.9	79.2±1.0	78.3±1.1	83.7±1.2
MIT Indoor Scene	34.1	36.3	37.9	38.6

doi:10.1371/journal.pone.0063385.t003

The discussions aforementioned only provide intuitive results. More strictly, we will verify the effectiveness of discriminative feature selection (DFS) via quantitative analysis. The prominent contribution of this work is that we can identify the discriminative local features from common ones in multiple images. Therefore, by utilizing these discriminative features, a discriminative codebook (DC) is consequently constructed for image-level histogram generation. The DC is different from typical codebook (TC) that in DC the codewords are the centers of the discriminative local features after Bayesian learning. On the other hand, in typical codebook, the codewords are clustered by using all the local SIFT features from the training images.

For image classification, in typical BoF framework, the image features (obtained after image encoding) are fed into a discriminative classifier, *e.g.* a Support Vector Machine (SVM) for final categorization. However, in this bi-class categorization task, if the discriminative codebook are utilized, we can even classify the images just by their feature saliency without training an extra classifier. The basic idea for feature-saliency-based bi-class categorization algorithm can be well illustrated by the following toy demo. We generate the discriminative codebook for the bi-classes categorization task in Fig. 4(a). After generating the codebook by only using the discriminative features, we assign the images to the codewords according to [20] and the visual results are provided in Fig. 5. It is obvious that in the Seaside histogram, more SIFT features are assigned to the first 100 codeword. Note that these 100 codewords are generated by clustering all the discriminative features from Seaside determined by our model. Therefore, by simply calculating the energy distribution of the histogram, the image can be accordingly classified.

In the previous paragraph, we have introduced the feature-saliency-classifier (FSC) for bi-classes image classification by using discriminative codebook. Of course, the same features can also be classified following the routine way of training an SVM. We will experimentally compare these two classifiers with the same features in Table 2. Besides, we also report the classification results on the features generated by using typical codebook. In the typical codebook, the codewords no longer contain class information and thus we cannot classify the images by exploiting feature saliency. They can only be classified by training an SVM. In this simulation, 1000 codewords are selected in the codebook. For training, 10 images in the SAR dataset are randomly selected as training samples. For the remaining five tests on natural images, 50 images in each category are selected as training samples. In each task, the experiments are randomly repeated for 10 times and the average classification accuracy and their standard deviations are tabulated in Table 2.

In the Table, the first and the second row report the classification results by using the DFS-codebook. The first row shows the classification result with FSC, and the second row shows classification accuracy with SVM classifier. In these two rows, the same features are fed into two different classifiers, *i.e.* the FSC and SVM. From the result, obviously, SVM has higher performance than that of FSC in five out of six tasks. There is no surprise to see such improvements because SVM is a strong classifier that requires extra training costs. FSC only vote for the classification by counting the feature saliency and requires no training. By reporting the results of FSC in this part, we just want to claim that the features learned after DFS itself have exhibited significant class attributes and they can even be easily classified by a classier as simple as FSC. From the result, it is also interesting to note that for the same features with DFS-codebook, the performance of FSC is comparable to the results of SVM.

The third row in the table shows the classification results by using typical codebook, which means that the codebook is generated by simply clustering all the local features in the images. It is obvious that the codewords in this codebook have no discriminative characters. By comparing the classification results in the second and third rows, it is found that by using the same classifier, the performance of DFS-codebook is much better than the typical codebook. This finding serves to verify that utilizing the differences among images actually helps recognition.

Results on Multi-class Categorization

In previous parts, the proposed algorithm is evaluated on bi-class classification task. In this part, we consider a more general case of applying the proposed model to improve the classification accuracy on multi-class tasks. For multi-class case, the learning procedures of discriminative feature selection are processed via a one vs. others paradigm. In a nutshell, we assume that there are m classes in total. To select the discriminative features for the p^{th} class, the features for the p^{th} class are regarded as positive features and the features from other $m-1$ classes are treated as negative ones. For the ease of calculation, both the positive and negative features are clustered into 4000 centers. After discriminative feature selection by the EM iterations, we only keep the discriminative features for the positive dataset, denoted as F_{p}, in our selection result. These features in F_{p} represent those that help to distinguish the images in the p^{th} class from others. These procedures are repeated for m times and the final codewords are clustered with all the features in $F_{\text{p}}, \forall p = 1...m$.

To verify the results, three datasets discussed above are all included in the experiment. In the SAR dataset, 10 images per category are used as training samples and the rest are test samples. In Fifteen Scene dataset, 100 images in each category are randomly selected as training samples and the rest are for test. In Indoor Scene dataset, we use the training and test samples provided in [22]. In the simulation, the multi-class SVM is applied and we strictly follow the implementations in some previous works [20,32] to conduct the experiments. In addition, we vary the number of codewords as 200, 400, 600, and 800 to investigate the robustness of the proposed method. Moreover, two benchmark feature coding algorithms, *i.e.* hard assignment [23] and kernel assignment [20] are respectively applied to the selected codewords for image-level feature generation. As indicated in [22], the categorization of indoor images is quite challenging and therefore we further exploit the $2-level$ spatial-pyramid [23] to improve the accuracy on MIT Indoor Scene dataset.

Fig. 6 shows the average classification accuracy of three datasets with regard to different sizes of codebook. It is shown that with the increase of the number of codewords, the classification accuracy is correspondingly improved. As expected, by using the same codebook, kernel assignment achieves higher performance than hard assignment. In Fig. 6, solid lines refer to categorization accuracies achieved from the codebooks generated by discriminative feature selection (DFS) using our proposed model, whereas the dashed lines represent those with respect to codebooks generated by simply clustering all features from multiple classes. It is indicated that by using discriminative feature selection (DFS) to generate the codebooks, the categorization accuracies on three datasets with regard to different sizes of the codebook are apparently enhanced.

It is worth noting that on SAR dataset which has only six categories, with DFS there is a considerable improvement on classification performance. In contrast, for the challenging Indoor Scene dataset with 67 categories, the enhancement in classification is less significant. We attribute this to the difficulty in selecting discriminative features from a large number of classes. Specifically, it is hard to separate discriminative features from the common features among as many as 67 classes, while it is comparatively easy to address 6 classes. But, even on these difficult task, our DFS method still improves the classification accuracy for 1.3% on the MIT Indoor Scene dataset with 800 codewords.

In the discussions aforementioned, we have verified the effectiveness of DFS on image categorization. Related image categorization work [23] has indicated that the classification accuracy could be significantly improved by using the pyramid methods. In order to achieve classification results comparable with other state-of-the-arts, we further follow the idea in [23] to conduct the classification with a 3-level pyramid. The number of codewords is fixed as 800. The experimental results are repeated for 10 times and the corresponding results (average accuracy ± std %) are reported in Table 3. In the indoor scene dataset, to make the result comparable with previous published paper, we directly use the training and testing list provided in [22] and the experiment is only implemented once.

In Table 3, it is indicated that by using $3-level$ spatial-pyramid and a larger number of codewords, the classification performance is significantly enhanced. On SAR dataset, the classification accuracy can achieve higher than 87%. Specifically, on Fifteen Scene dataset, the classification accuracy is as high as 83.7%, which is comparable with many state-of-the-arts. On the very challenging indoor scene dataset, we produce a categorization rate as high as 38.6% by the DFS method with soft assignment. The result is higher than some recent reported results by using the same training and testing samples [22].

Limitations of Study

Although there is much remains to be done, our work has introduced the preliminary idea of finding image differences by a generative probabilistic model. In other words, we have shown that discovering image differences among multiple classes indeed helps recognition tasks. Although the presented studies have yielded some preliminary findings, a few limitations still exist in the current work.

First, in this paper, the discovery of image differences is fully implemented in a computational statistical framework. Our work fully relies on a data-driven manner to identify the discriminative patterns. Therefore, the learning results are partially determined by what kind of local descriptors are used as the input of the model. In this work, we choose the most prevalent image descriptor, *i.e.* the SIFT feature, to represent the contents of images. However, describing image content by a single feature will be possibly insufficient. Therefore, there exists some artifacts on the discriminative feature selection results in Fig. 4. In our future works, we may consider to find the differences among images by using multiple descriptors.

Besides, with regard to the image classification accuracy, the performance of this method is not the highest reported one on some benchmarks [19]. In previous works for image categorization, multiple implementations have been adopted to improve the classification performances. For example, multiple descriptors were used in some systems with nonlinear classifier. The final classification results are voted by different discriminative machines. Therefore, there is no surprise to find some higher accuracy in these papers. In contrast, in this paper we only used one kind of descriptors with a single SVM classifier for image categorization. Even with such a simple implementation, after spatial pyramid enhancement [23], the classification results are very competitive with previous published results that used multiple approaches for image understanding. Meanwhile, we would like to emphasize that the core contribution of this work is not in the pursuit of the highest quantity on the classification accuracy. The aim of this paper is to reveal that differences among images indeed help recognition. Accordingly, the experiments designed in this work all focus on this prominent goal.

Conclusions

This paper proposes a generative model to interpret how different and common features are generated in images from multiple classes. The whole mathematical framework can be efficiently solved in an EM paradigm with very robust convergence. The experimental results from different perspectives, including visual effectiveness comparison, bi-class categorization and multi-class classifications, all verify that differences on images is very critical for improving the classification performances. Although we concern on the task of image categorization, the effectiveness of the proposed model is obviously beyond the scope discussed in this paper. The statistical model is a a general method to distinguish discriminative features from common ones. Therefore, it can be applied to a diversity of practical applications that involve discriminative feature selection.

Supporting Information

File S1 Derivations for the EM algorithm.
(PDF)

Author Contributions

Conceived and designed the experiments: YD QHD. Performed the experiments: YD YYZ YBL. Analyzed the data: YD QHD. Contributed reagents/materials/analysis tools: YBL. Wrote the paper: YD YYZ.

References

1. Haxby J, Ungerleider L, Horwitz B, Maisog J, Rapoport S, et al. (1996) Face encoding and recognition in the human brain. Proceedings of the National Academy of Sciences 93: 922–927.
2. De Renzi E, Scotti G, Spinnler H (1969) Perceptual and associative disorders of visual recognition. Neurology.
3. Frances Wang R, Simons D (1999) Active and passive scene recognition across views. Cognition 70: 191–210.
4. Schettino A, Loeys T, Bossi M, Pourtois G (2012) Valence-specific modulation in the accumulation of perceptual evidence prior to visual scene recognition. PLoS ONE 7: e38064.
5. Graziano M, Sigman M (2009) The spatial and temporal construction of confidence in the visual scene. PLoS ONE 4: e4909.
6. Diwadkar V, McNamara T (1997) Viewpoint dependence in scene recognition. Psychological Science 8: 302–307.
7. Huang W, Yin H (2012) On nonlinear dimensionality reduction for face recognition. Image and Vision Computing 30: 355–366.
8. Yin H, HuangW(2010) Adaptive nonlinear manifolds and their applications to pattern recognition. Information Sciences 180: 2649–2662.
9. Ben-Hur A, Ong C, Sonnenburg S, Schölkopf B, Rätsch G (2008) Support vector machines and kernels for computational biology. PLoS Computational Biology 4: e1000173.
10. Deng Y, Dai Q, Zhang Z (2011) Graph Laplace for occluded face completion and recognition. IEEE Transactions on Image Processing 20: 2329–2338.
11. Friedman J, Hastie T, Tibshirani R (2001) The Elements of Statistical Learning, volume 1. Springer Series in Statistics.
12. Fukunaga K (1990) Introduction to Statistical Pattern Recognition. Academic Pr.
13. Lafferty JD, McCallum A, Pereira FCN (2001) Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., ICML '01. pp.282–289. URL http://dl.acm.org/citation.cfm?id = 645530.655813.
14. Pan L, Chu W, Saragih J, De la Torre F, Xie M (2011) Fast and robust circular object detection with probabilistic pairwise voting. IEEE Signal Processing Letters 18: 639–642.
15. Deng Y, Dai Q, Wang R, Zhang Z (2012) Commute time guided transformation for feature extraction. Computer Vision and Image Understanding 116: 473–483.
16. Liu R, Lin Z, Su Z, Tang K (2010) Feature extraction by learning lorentzian metric tensor and its extensions. Pattern Recognition 43: 3298–3306.
17. Mairal J, Bach F, Ponce J, Sapiro G, Zisserman A (2008) Discriminative learned dictionaries for local image analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008. pp.1–8. doi:10.1109/CVPR.2008.4587652.
18. Jiang Z, Lin Z, Davis L (2011) Learning a discriminative dictionary for sparse coding via label consistent k-svd. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011. pp.1697–1704. doi:10.1109/CVPR.2011.5995354.
19. Tuytelaars T, Fritz M, Saenko K, Darrell T (2011) The NBNN kernel. In: 2011 IEEE International Conference on Computer Vision (ICCV). pp.1824–1831. doi:10.1109/ICCV.2011.6126449.
20. van Gemert JC, Veenman CJ, Smeulders AW, Geusebroek JM (2010) Visual word ambiguity. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32: 1271–1283.
21. Fei-Fei L, Perona P (2005) A bayesian hierarchical model for learning natural scene categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. IEEE, volume 2. pp.524–531.
22. Quattoni A, Torralba A (2009) Recognizing indoor scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009. IEEE. pp.413–420.
23. Svetlana L, Cordelia S, Jean P (2006) Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006. IEEE, volume 2. pp.2169–2178.
24. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. IEEE, volume 1. pp.886–893.
25. Libioulle C, Louis E, Hansoul S, Sandor C, Farnir F, et al. (2007) Novel Crohn disease locus identified by genome-wide association maps to a gene desert on 5p13. 1 and modulates expression of *ptger4*. PLoS Genetics 3: e58.
26. Brynedal B, Duvefelt K, Jonasdottir G, Roos I, Åkesson E, et al. (2007) Hla-a confers an hla-drb1 independent influence on the risk of multiple sclerosis. PLoS ONE 2: e664.
27. LaFramboise T, Weir B, Zhao X, Beroukhim R, Li C, et al. (2005) Allele-specific amplification in cancer revealed by snp array analysis. PLoS Computational Biology 1: e65.
28. Shoemaker B, Panchenko A (2007) Deciphering protein–protein interactions. part ii. computational methods to predict protein and domain interaction partners. PLoS Computational Biology 3: e43.
29. Borman S (2009) The expectation maximization algorithm-a short tutorial. Technical Report.

30. Deng Y, Dai Q, Liu R, Zhang Z, Hu S (2013) Low-rank structure learning via nonconvex heuristic recovery. IEEE Transactions on Neural Networks and Learning Systems 24: 383–396.

31. Bishop C (2006) Pattern Recognition and Machine Learning, volume 4. Springer New York.

32. Yang J, Yu K, Gong Y, Huang T (2009) Linear spatial pyramid matching using sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009. IEEE. pp.1794–1801.