Bio-Inspired Object Recognition Using Parameterized Metric Learning

Xiong Li, Bin Wang*, Yuncai Liu
Department of Automation, Shanghai Jiao Tong University
Shanghai 200240, China
[e-mail: binley.wang@gmail.com]
*Corresponding author: Bin Wang

Received December 11, 2012; revised January 25, 2013; revised March 2, 2013; accepted March 19, 2013; published April 30, 2013

Abstract

Computing global features based on local features using a bio-inspired framework has shown promising performance. However, for some tough applications with large intra-class variances, a single local feature is inadequate to represent all the attributes of the images. To integrate the complementary abilities of multiple local features, in this paper we have extended the efficacy of the bio-inspired framework, HMAX, to adapt heterogeneous features for global feature extraction. Given multiple global features, we propose an approach, designated as parameterized metric learning, for high dimensional feature fusion. The fusion parameters are solved by maximizing the canonical correlation with respect to the parameters. Experimental results show that our method achieves significant improvements over the benchmark bio-inspired framework, HMAX, and other related methods on the Caltech dataset, under varying numbers of training samples and feature elements.

Keywords: perceptual distance, parameterized metric learning, feature fusion, bio-inspired object recognition

This work was supported by National Basic Research Program of China 2011CB302203, China NSFC Key Program 60833009, NSFC Program 60975012.

http://dx.doi.org/10.3837/tiis.2013.04.012

1. Introduction

Motivated by research in human vision, machine learning and other relevant fields, object recognition has made rapid progress in recent years. Nevertheless, in comparison with a human being's vision system, which can distinguish approximately 30,000 categories with a few training samples, there is still a long, arduous way to go for future researches. In this paper, we focus on the specific branch of bio-inspired approaches [1,2,3,4,5,6,7,8], which has shown promising performance in a wide range of applications.

Research on human vision has developed a fundamental framework for object recognition, into which several computer vision models may be embedded. The representation of images can be very complex and highly diverse [9,10]. It has been argued [11], that categories are defined by their similarity to prototypes (i.e., specific image patches or other representations) rather than by lists of abstract quantities. In the prototype based framework, perceptual distances (or similarity measures) defined over prototypes rather than feature spaces, lie in the focus of researches. The primary advantages of this framework are twofold: (1) scaling the framework to a larger number of categories is quite straightforward, just by introducing enough prototypes and; (2) it allows designing the perceptual distance functions which are invariant to certain transformations or intra-class variations. Consequently, it is possible to train models with very few training samples. Serre and Poggio [2,3] proposed an hierarchical computational model (HMAX) for object recognition. The graphical illustration of an HMAX can be found in **Fig. 2**. HMAX is the modelling of the ventral stream of the primate visual cortex. The model is comprised of two types of layers, defined as S layers and C layers. S layers are the modelling of simple cells, which exhibit selectivity to orientation and scale, and can be modelled by Gabor filters [2,3,12]. C layers are the modelling of complex cells, which summarize the signals of different scales and orientations, and can be modelled by a maximization operation [3]. In the perspective of feature representation, the output of S layers are local features because they are the responses to image patches, rather than the whole image, while the output of the C layers are global features since the whole image is taken into account when computing each feature element. HMAX also can be considered as a framework that computes global features based on local features. HMAX is further extended [5,6] by considering more properties of the human vision system.

However, local features designed for specific vision tasks might fail under some extreme conditions, e.g., lighting and viewpoints. For instance, SIFT [6] which was originally designed to represent multi-level quantization images (e.g., 256 levels gray images) will fail on binary images. Also, even images from the same category have large intra-class variances, as shown in **Fig. 1**. Therefore, single local features might be not sufficient to represent all attributes of complex objects. It has been recently proposed [13,14] that multiple local feature representations could address the above problem under the distance function learning framework. Motivated by the capability of bio-inspired models, we extended HMAX to adapt multiple local features for global feature extraction, and to explore integrating global features using metric learning.

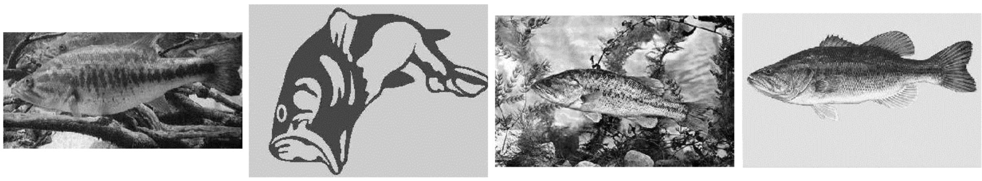

Fig. 1. Image samples from the Bash category of the Caltech 101 dataset. Great appearance variation appears in these images. The reasons leading to the variances are differences in enviroment, biology, morphology, etc. The second image loses most of the texture information and has a different pose than the others. The first and third images have complex and confusing backgrounds.

In this paper, we have generalized HMAX [2,3] to a global feature computation framework so that it can accommodate multiple local features, and we proposed a metric learning method to fuse high dimensional global features based on [1]. The framework for the proposed method is illustrated in **Fig. 3**. Although the generalization is based on an HMAX model, it also can be applied to other models [4,15] straightforwardly. The generalized model can adapt multiple local features and compute their corresponding global features. Then, a *parameterized metric learning* algorithm is developed to fuse these global features for object recognition. Essentially, this aligns the features at a metric level. For the features fusion task, a criterion defined as the *maximal kernel canonical correlation* [16,17] is leveraged to solve the metric learning problem. Our main contributions can be summarized into two points: (1) To extend the HMAX global features computation model to adapt multiple complementary local features, which greatly improves the capability of the system to recognize difficult categories, and, (2) to propose a metric learning method, parameterized metric learning, to make high dimensional feature fusion practical. Each set of metric weights is derived from the same template function, with a few parameters to be solved. In this way, the fusion model can be solved with a few training samples. We have also introduced the canonical correlation, to learn the metrics to fuse different global features together.

2. Related Works

This section reviews the related works, focusing on the HMAX [2,3] and its variants [5]. The HMAX models the ventral stream of the primate visual cortex, as an hierarchical structure for object recognition. The model is composed of two types of layers, S layers and C layers. S layers are the modelling of simple cells, which are selective to the signals of certain scales, orientations, etcetra, and can be implemented by Gabor filters [2,5]. In computations, S layers decompose signals into multiple channels, each of which is created by a filter selective to certain scales and orientations. C layers are the modeling of complex cells, which summarize the signals of different channels and can be implemented by a maximization operation or a pooling operation.

The signal in the HMAX streams as $S_1 \rightarrow C_1 \rightarrow S_2 \rightarrow C_2$, where S_1 and C_1 are the first simple cell and the first complex cell, respectively, and S_2 and C_2 are the second simple cell and the second complex cell, respectively. The structure of the HMAX is illustrated in **Fig. 2**. C_1 produces local features that are invariant to scaling and rotation, and C_2 computes global features by defining perceptual distance or similarity functions. Corresponding to the simple cells in the visual cortex, S layers improve the selectivity while C layers improve the invariance. Mutch and Lowe [5] extended the HMAX to consider sparsification and lateral inhibition, so that the features can benefit from the robustness of sparse representation. Serre and Poggio [18] proposed a feedforward and rapid recognition mechanism. Pinto and Cox [6]

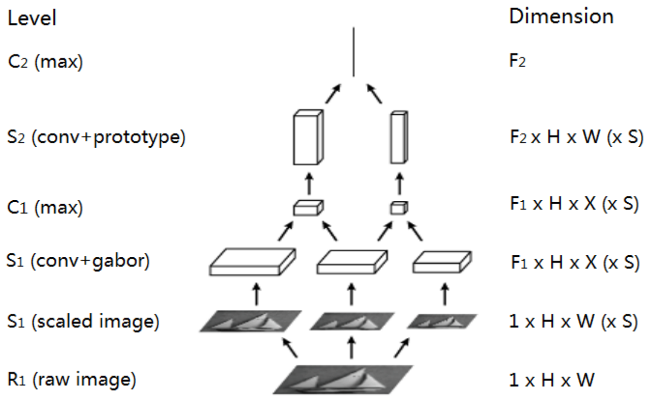

Fig. 2. Illustration of the framework of HMAX [3,5]. S indicates a simple cell and C indicates a complex cell. H and W are the height and width of the input image. S is the number of scales.

proposed a more biologically plausible, feature computing framework. This framework [4] is set to learn a perceptual distance function with a metric learning algorithm [19], which determines weights for feature elements. In concept, these algorithms [4,19] learn a transformation for the entire sample space.

The HMAX model and its variants follow the insight of Rosch [11] and share the following basic outline. Step 1, for an image, is the selection of a set of image patches. Step 2, for each patch, is the computation of a local feature, or C_1 response, and then the image is represented by a set of local features. In Step 3, a set of prototypes is learned from the local features, and in Step 4 a distance function is chosen or learned [4,15] from local features and prototypes. Such a distance function is known as a *perceptual distance*, and in Step 5, for an image represented by a set of local features, a set of distances is returned [3] (with respect to a set of prototypes, followed by the maximization operation) as the global feature, or C_2 response, of the input image.

3. Generalization of HMAX Perceptual Framework

We considered a typical, perception-inspired framework HMAX [2], with a structure as shown in **Fig. 2**. HMAX is composed of the following four steps: (1) Computing the C_1 response to a given image, which is referred to as the C_1 feature, (2) Learning the patch level prototypes from the C_1 features for the i-th prototype, (3) Computing its convolution over the C_1 feature of an image, the response for which is referred to as the i-th S_2 response, and (4) For the i-th S_2 response, computing the maximum response, and refering to it as the i-th element of the C_2 feature. In the following section, we proceed to generalize the HMAX based on the work of Li etal [1].

3.1. From the C1 Response to C1 Descriptors

To generalize the HMAX to accommodate multiple local features, we must first represent an image as a set of patches, and then treat the C_1 response to the image as a set of C_1 descriptors/local features of a set of patches. Then all the steps are updated accordingly, and other local features can be introduced into the framework by replacing the C_1 descriptor. In the next section, we describe how to compute the raw global features from the local features. The fusion scheme for global features will be introduced in Section 4.

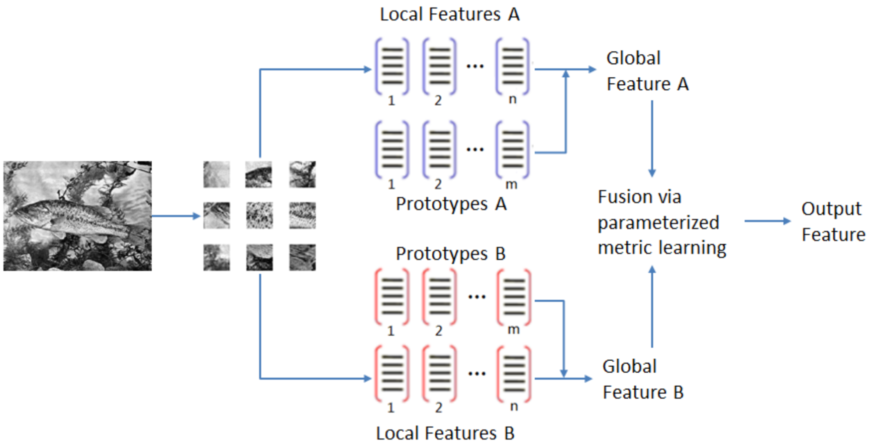

Fig. 3. Illustration of our proposed framework

3.2. Generalized Perceptual Distance

Let $\{P_i^n\}_{i=1}^K$ be a set of K patches extracted from the n-th image I^n, where the size of the patches can be different and the location of the patches can also be different (e.g., located by grids or interest point detectors). Let $\{\mathbf{c}_i^n\}_{i=1}^K$ be the local feature set corresponding to the patch set $\{P_i^n\}_{i=1}^K$, where the local feature $\mathbf{c}_i^n$ is correspondingly extracted from the patch P_i^n. In the learning procedure, we randomly selected a set of local features $\{\mathbf{c}_i^*\}_{i=1}^D$ from the local feature set of natural images as the prototypes. Then, for a specific type of local feature, the i-th element of the global feature $\mathbf{x}(I^n) \in R^D$ is defined as,

$$x_i^n(I^n) \overset{def}{=} \min_{c_j^n \in C(I^n)} \operatorname{dist}(\mathbf{c}_j^n, \mathbf{c}_i^*) \tag{1}$$

where function $\operatorname{dist}(\cdot,\cdot)$ is a distance function between local feature $\mathbf{c}_j^n$ and prototype $\mathbf{c}_i^*$; and the minimum distance can be regarded as an implementation of the maximal neural response. It is worth noting that only two descriptors from patches with the same size could be used to compute the perceptual distance. For the C_1 feature, the maximum neural response with respect to the C_1 prototypes corresponds to the shape tuning process of the visual cortex [2]. Note that x_i^n is the representation of patches with respect to the prototype $\mathbf{c}_i^*$.

The above extension allows the accommodation of the local features extracted from patches, such as C_1 [3], SIFT [12], shape context [20] and geometric blur [21]. It is interesting to note that C_1 and most of the other local features follow the scale space theory [22], and are invariant to rotation, scaling or affine translation. This fact indicates that these features can naturally work with C_1. In our work, two local features, C_1 and SIFT, are used in the extended model because of their complementarity. C_1 encodes rich contour and shape information, and SIFT encodes rich gradient information, complementarily. In the following sections, the C_1 based global feature is referred to as $\mathrm{C}_{2,}$ and the SIFT based global feature is referred to as SIFT_2. We then employed Euclidean distance and the normalized inner production as the perceptual distances for C_1 and SIFT, respectively.

In the extended model, as illustrated in Fig. 3, the final global feature is computed according to the following five steps: (1) Extract the patches from experimental images and arbitrary natural images, where the natural images are used for prototype learning, (2) Given a type of local feature, extract a set of features from the patches, (3) Learn prototypes from the local features of natural images, (4) Compute the global feature for an image, based on its local features and the learned prototypes, (5) Fuse the global features as the final global feature. Multiple types of raw global features can be computed by replacing the local feature in step 2. These steps are summarized in Algorithm 1.

Algorithm 1 Compute global feature

```
Input: a set of images $\{I^n\}_{n=1}^N$, and a specified local feature (e.g., $C_1$ or SIFT)
for $n=1$ to $N$ do
    extract $K$ patches $\{P_i^n\}_{i=1}^K$ from $I^n$
    extract $K$ local features $\{\mathbf{c}_i^n\}_{i=1}^K$ from $K$ patches $\{P_i^n\}_{i=1}^K$
end for
extract a set of local features $\{\mathbf{c}_i^*\}_{i=1}^D$ from natural images as the prototype set
for $n=1$ to $N$ do
    for $i=1$ to $D$ do
        $x_i^n(I^n) = \min_{c_j^n \in C(I^n)} d(\mathbf{c}_j^n, \mathbf{c}_i^*)$ (Eq. (1))
    end for
end for
Output: $\{\mathrm{x}^n\}_{n=1}^N$ where $\mathrm{x}^n = (x_1^n, \cdots, x_D^n)^T$
```

4. Parameterized Metric Learning for Feature Fusion

In the previous section, two global features, C_2 and SIFT$_{2,}$ are given, which are respectively based on C_1 and SIFT. However, global features derived from different local features or different perceptual distance functions have different metrics. Typically, a set of weights, each representing a feature, are learned for feature fusion. This method is not flexible enough, because the metric difference within a feature is ignored. Research [13] has shown that features fusion can benefit from subspace learning. However, applying this method to our problem is expensive, due to the high dimensional feature spaces.

Metric learning [4,19] is a popular method for feature fusion. It usually learns a metric for each dimension. This method is also very expensive for our problem because its computation cost increases as the numbers of dimensionality increase. In this section, we propose a novel metric learning method defined as *parameterized metric learning*, to deal with high dimensional feature fusion. A criterion known as maximal canonical correlation, is used to solve the metric weights.

4.1. Parameterization of Metrics

Regular metric learning methods [4,19] must determine a large number of independent weights. These methods are computationally expensive for high dimensional feature spaces. They tend to fail even when the number of training samples is relatively small. To make the metric learning practical for high dimensional feature space and few training samples, we propose parameterized metric learning for high dimensional feature fusion. The weights assigned to feature dimensions are derived from template functions, with few parameters. Consequently, we only have to determine a few function parameters instead of a large number of weights.

For a given image I^n, we suppose that similar feature elements correspond to similar prototypes. Therefore they share similar metrics, and should have similar weights. We use a continuous function $h(\theta) \in H$ as the 'template function' and assign the weights,

$$w_i = h(x_i^n, \theta) / x_i^n$$

to the global feature $\mathbf{x}^n(I^n) \in R^D$. Then the weighed feature $\overset{n}{\mathbf{x}}(I^n)$ can be formulated as:

$$\begin{aligned} \overset{n}{\mathbf{x}}(I^n) &= diag(w_1, \ldots, w_D)\mathbf{x}^n(I^n) \\ &= (h(x_1^n, \theta), \ldots, h(x_D^n, \theta))^T \\ &= h(\theta) \circ \mathbf{x}^n(I^n) \end{aligned} \tag{2}$$

where $\circ$ indicates that we apply function $h(\theta)$ over all elements of $\mathbf{x}^n(I^n)$. It suggests that the weighting feature, using the weights derived from template function h, is equal to applying h to the feature. Then the task of determining the weight set $\{w_i\}_{i=1}^D$ is converted to determine the parameters of the template function h. Weights derived from the same template function h, are nonlinearly dependent, because the number of free parameters of $\{w_i\}_{i=1}^D$ (equal to the number of parameters of h) is much smaller than D. This leads to a weak learning scheme. However, with the increasing capacity of template function h, the metric learning scheme will approach the regular metric learning. Similarly with the applications in [4,19], the proposed method can also be used for distance function learning, e.g., using maximal margin formulation for the triplets training set.

4.2. Solution via Canonical Correlation Maximization

Given the global features in Section 3, and the fusion method in Section 4.1, we proceed to solve the fusion problem in this section. The canonical correlations of within-class sets and between-class sets for discriminative learning have been explored, and some [24] have used canonical correlation analysis for feature fusion by determining pairs of projective matrices, given two candidate features. Here we used canonical correlation to determine the parameters of template functions, and the sequentially derived weights, instead of projective matrices [24], although sometimes weights can be regarded as a special projective matrix.

In this work, we used a canonical correlation maximization formulation [16,17] to solve the feature fusion problem. The reason is that two features representing the same object tend to reflect the same properties. Let $I = \{I^1, \cdots, I^N\}$ be a set of N images; $X_1 = (\mathbf{x}_1^1, \cdots, \mathbf{x}_1^N)^T$ and $X_2 = (\mathbf{x}_2^1, \cdots, \mathbf{x}_2^N)^T$ be two sets of global features, respectively computed from two types of local features using Eq. (1). Let $h(\theta_1)$, $h(\theta_2) \in H$ be two scalar functions parameterized by θ_1, θ_2. The two functions will be applied to two global features respectively. The canonical correlation of two weighted global features is,

$$\begin{aligned}\varphi(\theta_1,\theta_2,\alpha_1,\alpha_2) &= \mathrm{cov}(\alpha_1{}^T(h(\theta_1)\circ X_1),\alpha_2{}^T(h(\theta_2)\circ X_2)) \\ &= \alpha_1{}^T\mathrm{cov}(h(\theta_1)\circ X_1,h(\theta_2)\circ X_2)\,\alpha_2\end{aligned} \tag{3}$$

where cov is the canonical correlation; $h(\theta_1)\circ X_1$ represents applying transformation $h(\theta_1)$ on feature matrix X_1, as shown in Eq. (2); vectors $\alpha_1 \in R^{D_1}$ and $\alpha_2 \in R^{D_2}$ are respectively the combination coefficients of canonical correlation for X_1 and X_2. We determined optimum values of θ_1, θ_2, α_1, α_2 by maximizing Eq. (3) under a constraint:

$$\begin{aligned}&\max_{\theta_1,\theta_2,\alpha_1,\alpha_2} \alpha_1{}^T\mathrm{cov}\left(h(\theta_1)\circ X_1,h(\theta_2)\circ X_2\right)\alpha_2 \\ &s.t.\text{:}\ \mathrm{var}(\alpha_1{}^T(h(\theta_1)\circ X_1))=\mathrm{var}(\alpha_2{}^T(h(\theta_2)\circ X_2))=1\end{aligned}$$

For some simple template function $h(\theta)$, the constrained maximization problem can be solved using the Lagrange method,

$$L=\varphi(\theta_1,\theta_2,\alpha_1,\alpha_2)+\lambda_1(\mathrm{var}(\alpha_1{}^T(h(\theta_1)\circ X_1))-1)+\lambda_2(\mathrm{var}(\alpha_2{}^T(h(\theta_2)\circ X_2))-1)$$

where λ_1,λ_2 are Lagrange multipliers. In the following part, we will present a more general method, iterative optimization, to solve any template function.

First, we fix $h(\theta_1^{(t)}),h(\theta_2^{(t)})$ given by the iteration t and maximized Eq. (3) with respect to α_1 and α_2. Then the maximization problem of the iteration $t+1$ becomes,

$$\begin{aligned}&\max_{\alpha_1,\alpha_2}\alpha_1{}^T\mathrm{cov}(h(\theta_1^{(t)})\circ X_1,h(\theta_2^{(t)})\circ X_2)\,\alpha_2 \\ &s.t.\text{:}\ \alpha_1{}^T\mathrm{var}(h(\theta_1^{(t)})\circ X_1)\,\alpha_1=\alpha_2{}^T\mathrm{var}(h(\theta_2^{(t)})\circ X_2)\,\alpha_2=1\end{aligned} \tag{4}$$

This constrained maximization problem can be solved using the Lagrange method, leading to an eigenvalue problem. Denote $\Sigma_{12}=\mathrm{cov}(h(\theta_1^{(t)})\circ X_1,h(\theta_2^{(t)})\circ X_2)$, $M_1=\Sigma_{11}^{-1}\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$ and $M_2=\Sigma_{22}^{-1}\Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$. Then λ_1 is the common eigenvalue of matrixes M_1 and M_2, and α_1 is the eigenvector corresponding to λ_1.

Second, we fix $\alpha_1^{(t)}$ and $\alpha_2^{(t)}$ which are given by the iteration t, and maximized Eq. (3) with respect to θ_1,θ_2,

$$\begin{aligned}&\max_{\theta_1,\theta_2}\alpha_1^{(\mathbf{t})}{}^T\mathrm{cov}(h(\theta_1)\circ X_1,h(\theta_2)\circ X_2)\,\alpha_2^{(\mathbf{t})} \\ &s.t.\text{:}\ \alpha_1^{(\mathbf{t})}{}^T\mathrm{var}(h(\theta_1)\circ X_1)\,\alpha_1^{(\mathbf{t})}=\alpha_2^{(\mathbf{t})}{}^T\mathrm{var}(h(\theta_2)\circ X_2)\,\alpha_2^{(\mathbf{t})}=1\end{aligned} \tag{5}$$

The constrained optimization problem, in principle, can be solved using the Lagrange method. However, for most template functions (nonlinear functions), it is difficult to obtain the analytical solution of θ_1,θ_2. We note that, the differentials $\frac{\partial L}{\partial \theta_1}$ and $\frac{\partial L}{\partial \theta_2}$ are still available. Therefore, θ_1 and θ_2 can be solved using a numerical method, gradient descent, in this case. The overall learning procedure is summarized in Algorithm 2.

Once the parameter sets θ_1^* and θ_2^* are determined, two weighted global features can be computed by Eq. (2), leading to the final feature,

$$\mathbf{x}^n(I^n) = (h(\theta_1^*) \circ \mathbf{x}_1^n(I^n)^T, h(\theta_2^*) \circ \mathbf{x}_2^n(I^n)^T)^T$$

In the metric learning based fusion scheme, weighting of each feature element could be regarded as an adjusting process, with feedback signals in the visual cortex. The effectiveness of the feedback has been validated in previous research [25].

Algorithm 2 Parameterized metric learning for feature fusion

```
1: Input: two sets of global features $X_1 = \{x_1^n\}_{n=1}^N$ , $X_2 = \{x_2^n\}_{n=1}^N$ ; iteration number $T$
2: initialize parameters $\{\alpha_1^{(0)}, \alpha_2^{(0)}, \theta_1^{(0)}, \theta_2^{(0)}\}$
2: for $t = 1$ to $T$ do
3:     fix $\theta_1^{(t-1)}, \theta_2^{(t-1)}$ and solve $\alpha_1^{(t)}, \alpha_2^{(t)}$ using Eq. (4)
4:     fix $\alpha_1^{(t)}, \alpha_2^{(t)}$ and solve $\theta_1^{(t)}, \theta_2^{(t)}$ using Eq. (5)
5: end for
6: Output: $\{\alpha_1^*, \alpha_2^*, \theta_1^*, \theta_2^*\}$
```

4.3. Scaling to Multiple Features Fusion

The feature fusion method proposed in Section 4.2 is for two features. It is easy to scale the proposed method to multiple features fusion. For C sets of features $X_1, \cdots, X_C$, suggested by canonical correlation based methods [16,17], we have two fusion schemes. First the incremental scheme, in which we first fuse X_1, X_2 using Algorithm 2, and then fuse the fused feature with X_3, until all the features are fused. Second, we divide C sets of features into two groups, and fuse the two groups of features using Algorithm 2.

5. Experiments

Object classification experiments with the fused global feature are performed on the Caltech 101, to show the advantage of the extended model and the fusion scheme, and to examine the stability of fused features under varying numbers of samples and feature elements. To evaluate the proposed method, HMAX is chosen as the benchmark system because it provides the basic framework for our method.

5.1 Dataset and Experimental Setup

Caltech 101 contains 101 categories and 9,146 images in total. The number of images in each category varies from 40 to 800, and most categories have about 50 images. To speed up feature computation, all the images are normalized to gray images, with 140 pixels maximum and a fixed aspect ratio.

There are several methods developed to extract patches, interest point detectors and grids. Interest region detectors include MSER [26], Harris-Affine and Hessian-Affine [27] which can be embedded into our framework. According to the comparison studies [28], we selected the Hessian-Affine as the interest region detector for the first experiment, and selected grid regions for the second experiments.

Table. 1 Accuracy comparison of four types of features. See the details of these features in the text. The number of positive training samples, negative training samples, positive testing samples and negative testing samples are 30, 50, 50 and 50, respectively.

Data set	C_2[3]	$SIFT_2$	C_2+$SIFT_2$	slf-C_2[5]	V1-like[6]	Ours
Butterfly	80.92	82.20	85.15	83.55	85.97	**88.79**
Brain	81.12	83.47	84.58	84.61	85.74	**88.33**
Bonsai	79.69	80.05	81.30	82.67	83.55	**86.81**
Chandelier	77.83	78.97	78.91	79.54	80.57	**82.81**
Car-side	97.37	97.54	97.91	97.04	98.23	**99.29**
Airplanes	96.74	97.03	97.35	97.68	**98.14**	98.00
Buddha	79.47	82.53	83.49	82.96	83.98	**87.29**
Scorpion	77.54	79.41	80.58	81.09	**84.47**	84.38

For a candidate image, patches with the sizes of 4×4, 8×8, 12×12 and 16×16 are extracted from all the interest regions respectively. The C_1 descriptors are constructed for each patch, while SIFT descriptors are constructed for 12×12 and 16×16 patches. For prototype learning, 500 patches per size are randomly extracted. Although descriptors can be extracted for all size of patches, descriptors from 12×12 and 16×16 patches work well. Then two sets of prototypes are learnt from natural images for C_1 and SIFT, respectively.

In all experiments, we use C_1 and SIFT as the local features for their complementarity, and use Euclidean distance and the normalized inner production as the perceptual distances for C_1 and SIFT, respectively. We refer to the corresponding global features as C_2 and $SIFT_2$. See Section 3.2 for details.

5.2 One-vs-Rest over a Subset of Caltech

In this section, we developed a set of controlled experiments to verify the behaviors of the proposed methods. We deliberately chose eight difficult categories, and the background category, preferring those challenging categories of images taken under extreme light, point of view and mutative poses. The size of samples ranged from 80 to 800. In the experiments, the size of the positive training set and the length of the global features were varied. The numbers of negative training samples, positive test samples and negative test samples were set to 50 each. We used linear SVM as the classifier.

First, we evaluated the performance of six types of features: C_2 (2000 elements), $SIFT_2$ (2000 elements), the concatenation of C_2 and $SIFT_2$ (1600 C_2 elements and 400 $SIFT_2$ elements, referred to as C_2+$SIFT_2$), the fusion of C_2 and $SIFT_2$ (1600 weighted C_2 elements and 400 weighted $SIFT_2$ elements), slf-C_2 [5] and V_1-like [6]. We randomly selected 30 sets of samples and 20 subsets of features for experiments with 30×20 rounds in total. As shown in **Table 1**, all features show high accuracy for the Car-side and Airplanes, and relative low accuracy for other categories. This is because the images of Car-side or Airplanes have small variance and similar appearance, even though they are taken from different conditions. For all eight categories, the C_2+ $SIFT_2$ feature outperforms the C_2 feature by about 0.6 to 4.2%. On the other hand, it has been shown [3] that when increasing the number of feature elements, it is hard to improve the performance when the number is greater than 1,000. This supports the idea that fusing features computed from complementary local descriptors would be helpful. Compared with C_2 features, our fusion scheme reaches an improvement of about 1.3 to 1.9% for Car-side and Airplanes, and about 5.0 to 7.9% for the other categories. When compared with two state-of-the-art approaches, e.g., slf-C_2 [5] and V_1-like [6], the proposed approach also exhibits superior performance in most cases.

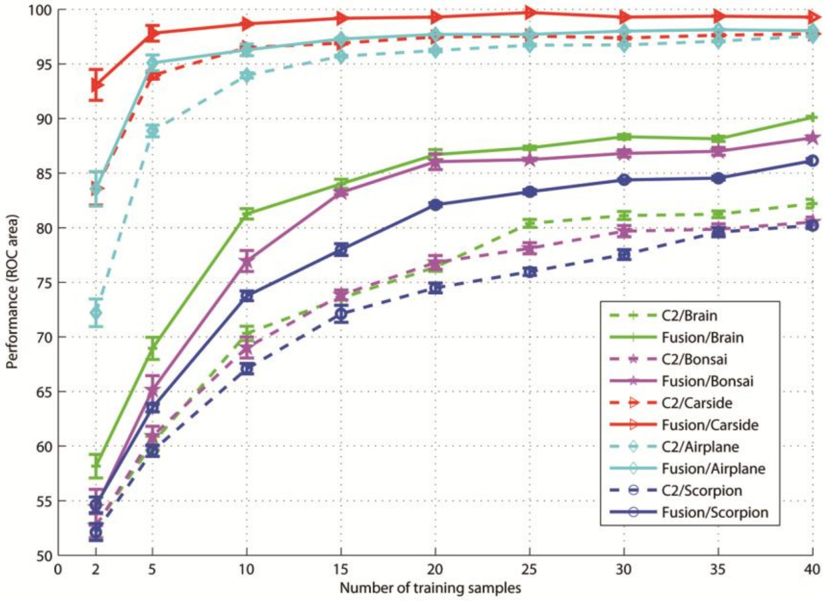

Fig. 4. Accuracy comparison of C_2 features (2,000 elements) and the proposed method (the fusion of 1,600 C_2 elements and 400 $SIFT_2$ elements) on the Caltech 8 for varying numbers of training examples.

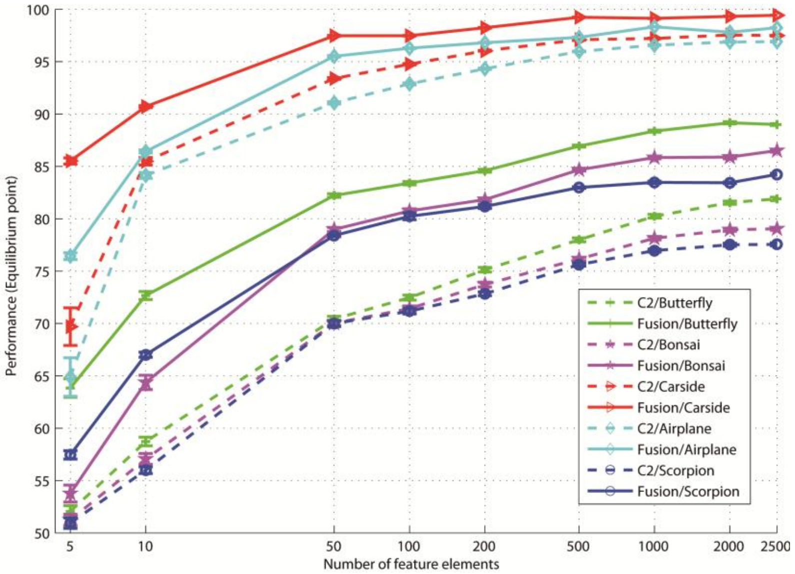

Fig. 5. Accuracy comparison of C_2 features and the fusion feature (75 % C_2 and 25 % $SIFT_2$ elements) on the Caltech 8 for varying numbers of feature elements.

To validate the robustness of the fusion scheme with respect to the number of training images, we varied the number of positive training images. Fig. 4 shows the results for 5 categories, where our fusion scheme outperforms C_2 in all five categories. For Car-side and Airplanes, the fusion scheme with five positive training images, achieves satisfied accuracies of about 97.8 and 95.1 %, with improvements of about 3.9 and 9.2 % respectively. For the other three categories, the fusion scheme with 20 positive training images reaches a significant accuracy of more than 83.1 %, while the accuracy of C_2 features is under 77.2 %. It also outperforms C_2 by about 5.1 to 7.9 %, when the number of positive training images is more than 20.

To validate the effectiveness of the proposed method with the number of feature elements, we performed a series of experiments by varying the the number of feature elements from 2 to 2,500. As shown in **Fig. 5**, the fusion scheme outperforms C_2 features for all numbers of elements. For categories of Car-side and Airplanes, the fusion scheme with 50 elements reached a satisfied accuracy of about 97.5 and 95.5 %, with improvements of about 4.0 and 4.5 %. For the other three categories, the fusion scheme with 100 elements reaches a significant accuracy exceeding 80 %, when the accuracy of the C_2 feature is no more than 72.5 %. For the settings of 50 or more feature elements, the fusion scheme outperforms C_2 features by at least 5.9 %, especially the 11.8 % for the Butterfly category.

5.3 Multiclass Recognition over Caltech 101

In the above section, the behaviors of the proposed methods are verified using a set of controlled experiments, i.e., varying a factor and fixing the other factors. In this section, we further evaluate their ability in real recognition tasks. We tested the proposed method and five related methods, on the Caltech 101 dataset. A typical setting [29] for the Caltech 101 dataset is used in this experiment. For each round of tests, we randomly selected 15 images from each category for training, and another 15 images for testing. To verify the robustness of the proposed approach to classifiers, we used three typical classifiers, SVM with linear kernel, SVM with RBF kernel, and multiple kernel learning (MKL). For SVM and MKL, we used the default parameters suggested by libsvm, [30] and [29] respectively**.** To ensure the results are comparable, we used its released splits of datasets, i.e., the UCSD-MIT Caltech-101-MKL Dataset [29].

Table. 2 The comparison evaluation of five features: SIFT [12], $SIFT_2$, C_2 [3], slf-C_2 [5], V1-like [6] and our proposed method on Caltech 101 dataset. The numbers of training samples and test samples per category are set to 15, respectively.

Classifier	SIFT [12]	$SIFT_2$	C_2 [3]	slf-C_2[5]	V1-like[6]	Ours
SVM(linear)	40.51	44.28	42.53	46.57	47.12	**48.52**
SVM(RBF)	42.83	44.34	41.98	47.05	48.89	**50.36**
MKL	42.75[29]	46.62	45.79	49.54[29]	50.83[29]	**52.18**

The results were averaged over 20 rounds of tests, and are reported in **Table 2**. As shown in **Table 2**, our method outperforms five other methods, for three different classifiers, which indicates that: (1) our method integrates the ability and information of C_2 elements and $SIFT_2$ elements, and (2) our method is robust for classifiers. We also find that $SIFT_2$ outperforms SIFT significantly, which verifies the effectiveness of the perception computation framework generalized in Section 3. Though slf-C_2 [5] and V1-like [6] outperform C_2 and $SIFT_2$ on all classifiers, they are inferior to our fusion method. Further, we note that MKL outperforms linear SVM and RFB-SVM on most features. This fact is consistent with experiments in other research [29]. The improvement of our method over other methods might result from two aspects. First, the fusion scheme eliminates the metric difference between features, and second, the underling methodology and theory (simple cells and complex cells) of SIFT features, are similar to C_1, therefore the resulting C_2 and $SIFT_2$ are compatible.

In the fusion scheme, the optimization process of Eq. (3) consumes much more time than the other steps. Using Gaussian function as the template function, solving Eq. (3) using gradient descent on a PC takes about 40 seconds per 80 training images. When the size of dataset grows, the computation complexity mainly depends on maximization of R^N. On the other hand, the patches that represent the candidate image are extracted from overlapping grids, [3] instead of interest regions, so Eq. (1) takes more time to compute the global features, but yields better performance.

6. Conclusions

In this paper, we extended a bio-inspired framework, HMAX, to adapt multiple local features to compute multiple global features. The point of the extension is the perception distance with respect to a set of prototypes. Then, we proposed parameterized metric learning for high dimensional features fusion. The metric learning model is solved through canonical correlation maximization formulation, producing final features. Experiments on the Caltech dataset show significant improvements over HMAX and other related methods, using settings with varying numbers of training images and feature elements, which also confirm the effectiveness and stability of the proposed method. The fusion scheme, however, achieves this performance at the cost of computational complexity, which will be a future research topic of this model.

References

[1] X. Li, X. Zhao, Y. Fu, and Y. Liu, “Weak metric learning for feature fusion towards perception-inspired object recognition,” in *Proc. of International Conference on MultiMedia Modeling,* pp. 273–283, 2010. Article (CrossRef Link)

[2] M. Riesenhuber and T. Poggio, “Hierarchical models of object recognition in cortex,” *Nature Neuroscience*, vol. 2, pp. 1019–1025, 1999. Article (CrossRef Link)

[3] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, “Robust object recognition with cortex-like mechanisms,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007. Article (CrossRef Link)

[4] A. Frome, Y. Singer, and J. Malik, “Image retrieval and classification using local distance functions,” *NIPS*, 2007.

[5] J. Mutch and D. Lowe, “Object class recognition and localization using sparse features with limited receptive fields,” *International Journal of Computer Vision*, vol. 80, no. 1, pp. 45–57, 2008. Article (CrossRef Link)

[6] N. Pinto and D. Cox, “High-throughput-derived biologically-inspired features for unconstrained face recognition,” *Image and Vision Computing*, vol. 30, no. 3, pp. 159-168, 2012. Article (CrossRef Link)

[7] K. Rajaei, S. Khaligh-Razavi, M. Ghodrati, R. Ebrahimpour, and M.E.S.A Abadi, “A stable biologically motivated learning mechanism for visual feature extraction to handle facial categorization,” *PLOS one*, vol. 7, 2012. Article (CrossRef Link)

[8] A.A. Maashri, M. Debole, M. Cotter, N. Chandramoorthy, Y. Xiao, V. Narayanan, and C. Chakrabarti, “Accelerating neuromorphic vision algorithms for recognition,” in *Proc. of Annual Design Automation Conference*, 2012. Article (CrossRef Link)

[9] J.Z. Leibo, J. Mutch and T. Poggio, “Why the brain separates face recognition from object recognition,” *NIPS*, 2011.

[10] B. Timothy, T. Konkle, A. George and A. Oliva, “Real-world objects are not represented as bound units: independent forgetting of different object details from visual memory,” *Journal of Experimental Psychology*, 2012.

[11] E. Rosch, “Natural Categories,” *Cognitive Psychology*, vol. 4, no. 3, pp. 328–350, 1973. Article (CrossRef Link)

[12] D. Lowe, "Object recognition from local scale-invariant features," *ICCV*, pp. 1150–1157, 1999. Article (CrossRef Link)

[13] Y. Fu, L. Cao, G. Guo and T. S. Huang, "Multiple feature fusion by subspace learning," *CIVR*, pp. 127–134, 2008. Article (CrossRef Link)

[14] Y. Lin, T. Liu and C. Fuh, "Dimensionality Reduction for Data in Multiple Feature Representations," *NIPS*, 2009.

[15] H. Zhang, A. Berg, M. Maire and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition," *CVPR*, 2006. Article (CrossRef Link)

[16] P. Lai and C. Fyfe, "Kernel and nonlinear canonical correlation analysis," *International Journal of Neural Systems*, vol. 10, no. 5, pp. 365–378, 2000.

[17] D. Hardoon, S. Szedmak and J. Shawe-Taylor, "Canonical correlation analysis: an overview with application to learning methods," *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004. Article (CrossRef Link)

[18] T. Serre, A. Oliva and T. Poggio, "A feedforward architecture accounts for rapid categorization," in *Proc. of the National Academy of Science*, vol. 104, no. 15, pp. 6424–6429, 2007. Article (CrossRef Link)

[19] M. Schultz and T. Joachims, "Learning a distance metric from relative comparisons," *NIPS*, 2004.

[20] S. Belongie, J. Malik and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 509–522, 2002. Article (CrossRef Link)

[21] A. Berg and J. Malik, "Geometric blur and template matching," *CVPR*, 2001. Article (CrossRef Link)

[22] T. Lindeberg and Scale-space, "A framework for handling image structures at multiple scales," *European Organization for Nuclear Research-Reports-CERN*, pp. 27–38, 1996.

[23] T. Kim, J. Kittler and R. Cipolla, "Discriminative learning and recognition of image set classes using canonical correlations," *IEEE Transactions on Pattern Analysis and Ma-chine Intelligence*, vol. 29, no. 6, pp. 1005–1018, 2007. Article (CrossRef Link)

[24] Q. Sun, S. Zeng, Y. Liu, P. Heng and D. Xia, "A new method of feature fusion and its application in image recognition," *Pattern Recognition*, vol. 38, no. 12, pp. 2437–2448, 2005. Article (CrossRef Link)

[25] S. Karayev, M. Fritz, S. Fidler and T. Darrell, "A probabilistic model for recursive factorized image features," *CVPR*, 2011. Article (CrossRef Link)

[26] J. Matas, O. Chum, M. Urban and T. Pajdla, "Robust wide-baseline stereo from maximal-ly stable extremal regions," *Image and Vision Computing*, vol. 22, no. 10, pp. 761–767, 2004. Article (CrossRef Link)

[27] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," *ECCV*, pp. 128–142, 2002. Article (CrossRef Link)

[28] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1, pp. 43–72, 2005. Article (CrossRef Link)

[29] N. Pinto, http://mkl.ucsd.edu/dataset/ucsd-mit-caltech-101-mkl-dataset.

[30] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1-27, 2011. Article (CrossRef Link)

Xiong Li received his BE degree in Aircraft Power Engineering from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2004; and the MS degree in pattern recognition and intelligent system from Southeast University, Nanjing, China. He currently is a PhD candidate at Department of Automation, Shanghai Jiao Tong University, Shanghai, China. His research interests include probabisltic graphical models and hybrid generative discriminative learning.

Bin Wang received her BE degree in information and computional science from Shandong Normal University, Ji'nan, China, in 2006; and the MS degree in applied mathmatics from University of Science and Technology Beijing, Beijing, China. She is currently a PhD candidate at Department of Automation, Shanghai Jiao Tong University, Shanghai, China. Her research interests include computer vision, machine learning, image processing, multimedia analysis.

Yuncai Liu received the Ph.D. degree in the Department of Electrical and Computer Science Engineering in 1990 from the University of Illinois at Urbana-Champaign (UIUC), and worked as an associate researcher at the Beckman Institute of Science and Technology from 1990 to 1991. Since 1991, he had been a system consultant and then a chief consultant of research in Sumitomo Electric Industries, Ltd., Japan. In October 2000, he jointed the Shanghai Jiao Tong University as a distinguished professor. His research interests are in image processing and computer vision.