Efficient Discriminative Local Learning for Object Recognition

Yen-Yu Lin[1,2], Jyun-Fan Tsai[1], and Tyng-Luh Liu[1]

[1]Institute of Information Science, Academia Sinica, Taiwan; [2]Dept. of CSIE, National Taiwan University, Taiwan

1. Summary

- We address two unfavorable issues of local learning, i.e., high risk of overfitting and heavy computational cost, and present an efficient boosting algorithm to learn sample-specific local classifiers for object category recognition.
- Our approach
  - We cast the multiple, independent training processes of local classifiers as a correlative multi-task learning problem.
  - We establish a parametric space where these local classifiers lie and spread as a manifold-like structure.
  - By designing a new multi-task boosting algorithm, the local classifiers are obtained by completing the manifold embedding.
  - The algorithm carries out incremental multiple kernel learning.

2. Background

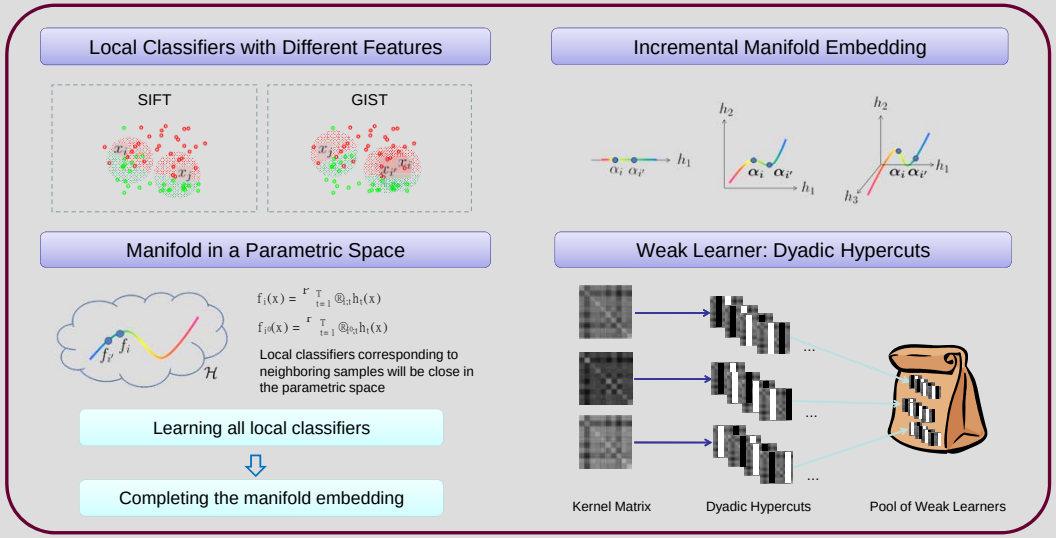
- Local learning and multi-task learning are the two key components of the proposed approach.

2.1 Local Learning

- Instead of a global model for the whole dataset, local learning employs multiple local models, each of which is optimized for a specific subset of data.
- Local learning is effective in tackling the difficulties caused by large intra-class variations in object recognition.
- Our method is to learn sample-specific local classifiers.
- Two issues should be considered:
  - High risk of overfitting: Each local classifier is learned with a small number of training data.
  - Heavy computational cost: Numerous local classifiers need to be learned.

2.2 Multi-task Learning

- Multi-task learning: Investigating related tasks simultaneously often achieves a considerable performance gain.
- By considering the learning of each local classifier as a task, accomplishing all the tasks jointly can be formulated as a multi-task learning problem.
- Why multi-task learning?
  - Tasks of learning local classifiers are highly correlative.
  - Training data of each task are insufficient.
  - Input/Output domains are the same.
- Proper regularization: The extra knowledge from other tasks benefits the completion of a specific task.
- Redundancy elimination: The information redundancy among all tasks can be appropriately modeled and eliminated, if these tasks are investigated simultaneously.



3. Problem Definition

- Our goal is to learn the sample-specific local classifiers $\{f_n\}_{n=1}^N$ for a given dataset $S = \{(x_n, y_n)\}_{n=1}^N$.
- Each local classifier f_i is designed to best discriminate data in the neighborhood of x_i, i.e., $S_i = \{x_n, y_n, w_{i,n}\}_{n=1}^N$ with

$$w_{i,n} = \begin{cases} 1=C; & \text{if } x_n \in C\text{-NN of } x_i; \\ 0; & \text{otherwise.} \end{cases}$$

3.1 Learning with Multiple Kernels

- In complex recognition problems, adopting multiple descriptors is critical for improving performances.
- Kernel as a unified feature representation:
  - Represent the pair-wise relationships among data under each descriptor by a kernel matrix.
- Data access can be conveniently done by referencing these kernels, without the trouble in resolving the diversity of descriptor forms.

3.2 Our Formulation

- Assume each local classifier is a boosted one.
- We model the relatedness among these local classifiers by considering that they share a common set of weak learners, i.e.,

$$f_i(x) = \sum_{t=1}^{T} \beta_{i,t} h_t(x); \quad \text{for } i = 1, 2, \ldots, N$$

  - All the weak learners $\{h_t\}_{t=1}^T$ are commonly shared.
  - The ensemble coefficients $\{\beta_{i,t}\}_{i=1,t=1}^{N,T}$ are task-dependent.
- All the local classifiers will be learned jointly.

4. The Proposed Approach

- A new multi-task boosting algorithm is developed to learn all the local classifiers jointly.

4.1 Design of Weak Learners

- We adopt dyadic hypercuts [Moghaddam & Shakhnarovich, NIPS'02] as the weak learners.
- A dyadic hypercut is specified by a kernel and a pair of training samples of opposite labels:

$$h(x) = \text{sign}(k(x_p, x) - k(x_n, x) - \mu):$$

- Dyadic hypercuts capture useful information in the kernel.
- Via taking the union of dyadic hypercuts generated by all the kernels, our boosting algorithm can learn a classifier by using information from multiple kernels.

4.2 Multi-task Boosting

- The loss function considers all the tasks:
  $\sum_{i=1}^{N} L(f_i, S_i)$, where $L(f_i, S_i) = \sum_{n=1}^{N} w_{i,n} \exp(-y_n f_i(x_n))$:
- A boosting algorithm via gradient descent.
- At iteration t, the weak learner shared across tasks is derived by

$$h_t = \arg\min_h \sum_{i=1}^{N} L(f_i + h, S_i):$$

- The task-dependent ensemble weight is determined by

$$\beta_{i,t} = \arg\min_{\beta} L(f_i + \beta h_t, S_i), \text{ for } i = 1, 2, \ldots, N:$$

- It carries out multiple kernel learning in an incremental manner.

5. Experimental Results

- The proposed approach is evaluated on two benchmark datasets: Caltech-101 and Pascal VOC 2007.

5.1 Caltech-101

- The Caltech-101 image dataset consists of 101 object categories and one additional class of background images.
- We randomly pick thirty images from each category. Fifteen of them are used for training, and the rest are for testing.
- We implement ten kinds of image descriptors that result in the following ten base kernel matrices:
  - GB-Dist / GB: Based on geometric blur descriptor.
  - SIFT-Dist / SIFT- SPM: Based on the SIFT descriptor.
  - SS-Dist / SS- SPM: Based on the self-similarity descriptor.
  - C2-SWP / C2-ML: Based on biologically inspired features.
  - PHOG: Based on PHOG descriptor.
  - GIST: Based on GIST descriptor.
- Recognition rates with a single kernel

	1-NN	SVM	AdaBoost	Ours
GB-Dist	42.4 ± 1.3	61.4 ± 0.8	61.5 ± 0.7	$\mathbf{63.2 \pm 0.8}$
GB	37.4 ± 0.9	57.6 ± 1.0	58.5 ± 0.7	$\mathbf{59.7 \pm 1.2}$
SIFT-Dist	49.6 ± 0.8	$\mathbf{58.7 \pm 0.9}$	57.5 ± 1.0	57.8 ± 0.7
SIFT-SPM	48.8 ± 0.7	$\mathbf{56.1 \pm 0.9}$	53.3 ± 0.8	55.2 ± 0.7
SS-Dist	31.7 ± 1.4	53.4 ± 1.0	56.1 ± 0.7	$\mathbf{56.3 \pm 0.9}$
SS-SPM	41.7 ± 0.9	53.9 ± 0.9	55.5 ± 0.7	$\mathbf{56.9 \pm 1.3}$
C2-SWP	22.0 ± 0.8	$\mathbf{28.3 \pm 0.8}$	26.0 ± 0.9	26.9 ± 1.0
C2-ML	37.7 ± 0.7	46.3 ± 1.0	44.8 ± 0.9	$\mathbf{46.6 \pm 0.6}$
PHOG	27.7 ± 1.0	$\mathbf{43.9 \pm 0.8}$	41.3 ± 1.0	42.9 ± 0.8
GIST	36.8 ± 1.1	48.7 ± 0.8	49.1 ± 1.0	$\mathbf{51.2 \pm 0.8}$

- Recognition rates and training time with multiple kernels

	SimpleMKL	AdaBoost (local)	Ours
All	74.3 ± 1.2 3.26×10^2 sec.	74.6 ± 1.3 1.87×10^5 sec.	$\mathbf{75.8 \pm 1.1}$ 3.92×10^3 sec.

5.2 Pascal VOC 2007

- We learn the classifiers with the Train+Val set, and evaluate the performance on the Test set.
- Six image descriptors are used for the dataset:
  - SIFT / GB / SS / GIST / C2-ML: As what described above.
  - TC-SIFT: Apply the SIFT descriptor to RGB channels separately.
- Three kinds of spatial pyramids are considered:
  - 1 x 1 (whole image), 2 x 2 (image quarters), and 1 x 3 (horizontal bars).
- The average precisions are reported as follows

	avg.	Aero.	Bicy.	Bird	Boat	Bott.	Bus	Car	Cat
INRIA	59.4	77.5	63.6	56.1	**71.9**	**33.1**	60.6	78.0	**58.8**
XRCE	57.5	72.3	57.5	53.2	68.9	28.5	57.5	75.4	50.3
TKK	51.7	71.4	51.7	48.5	63.4	27.3	49.9	70.1	51.2
van Gemert *et al.* [31]	**60.5**	**80.4**	**64.9**	**57.0**	69.1	24.6	**65.8**	**78.2**	54.3
SimpleMKL	57.3	74.1	62.7	48.7	66.9	29.1	62.6	75.0	56.9
Ours	59.3	76.5	64.6	51.8	68.3	32.2	61.3	77.5	57.8

Chair	Cow	Table	Dog	Horse	Moto.	Pers.	Plant	Sheep	Sofa	Train	Tv
53.5	42.6	54.9	45.8	77.5	64.0	85.9	36.3	44.7	50.6	79.2	53.2
52.2	39.0	46.8	45.3	75.7	58.5	84.0	32.6	39.7	50.9	75.1	49.5
51.7	32.3	46.3	41.5	72.6	60.2	82.2	31.7	30.1	39.2	71.1	41.0
56.9	42.4	53.7	**47.0**	**81.5**	65.6	**87.9**	**38.3**	**52.3**	53.9	**83.2**	53.3
54.5	42.7	54.8	44.2	76.3	**65.8**	83.6	28.7	42.5	51.5	74.7	50.9
56.3	**43.5**	**58.8**	44.8	78.4	65.2	85.4	30.4	47.7	**54.6**	76.4	**54.6**