Cogn Comput (2018) 10:49–61
DOI 10.1007/s12559-017-9489-x

A Novel Manifold Regularized Online Semi-supervised Learning Model

Shuguang Ding[1] · Xuanyang Xi[2] · Zhiyong Liu[2,3,4] · Hong Qiao[2,3,4] · Bo Zhang[1]

Received: 6 January 2017 / Accepted: 18 July 2017 / Published online: 2 August 2017

Abstract In the process of human learning, training samples are often obtained successively. Therefore, many human learning tasks exhibit online and semi-supervision characteristics, that is, the observations arrive in sequence and the corresponding labels are presented very sporadically. In this paper, we propose a novel manifold regularized model in a reproducing kernel Hilbert space (RKHS) to solve the online semi-supervised learning (OS^2L) problems. The proposed algorithm, named Model-Based Online Manifold Regularization (MOMR), is derived by solving a constrained optimization problem. Different from the stochastic gradient algorithm used for solving the online version of the primal problem of Laplacian support vector machine (LapSVM), the proposed algorithm can obtain an exact solution iteratively by solving its Lagrange dual problem. Meanwhile, to improve the computational efficiency, a fast algorithm is presented by introducing an approximate technique to compute the derivative of the manifold term in the proposed model. Furthermore, several buffering strategies are introduced to improve the scalability of the proposed algorithms and theoretical results show the reliability of the proposed algorithms. Finally, the proposed algorithms are experimentally shown to have a comparable performance to the standard batch manifold regularization algorithm.

Keywords Human learning · Manifold regularization · Online semi-supervised learning · Lagrange dual problem

✉ Zhiyong Liu
zhiyong.liu@ia.ac.cn

1 LSEC and Institute of Applied Mathematics, AMSS, Chinese Academy of Sciences, Beijing 100190, China

2 State Key Lab of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

3 CAS Centre for Excellence in Brain Science and Intelligence Technology (CEBSIT), Shanghai 200031, China

4 Cloud Computing Center, Chinese Academy of Sciences, DongGuan, GuangDong 523808, China

Introduction

Online learning techniques get great progress in recent years [1–10]. In general, online learning has several characteristics: (1) the samples arrive sequentially in a stream and only one new sample is available in each online learning round; (2) the label of the new arrived sample is predicted by the current classifier and the true label of the sample is revealed; (3) when a new sample is misclassified, the classifier should be updated in time to improve its generalization ability; and (4) the classifier can be updated without re-training all the visible samples.

In literature, much attention has been put on online supervised learning, e.g., [11–14], that is, the true labels are available in the online training process. However, in practice, we frequently face online semi-supervised learning problems [15–17], such as the human categorization problem. In [16], Zhu et al. designed a series of experiments to demonstrate that the human learning behavior is closely related to the semi-supervised learning pattern. Furthermore, Gibson et. al. [18] applied the learned semi-supervised model to human learning tasks. In human learning, learners can incrementally learn the classes of various objects from the surrounding environment, where only a few objects are labeled by a knowledgeable source. This scenario can be actually regarded as online semi-supervised learning, that is,

the label of a new arrived sample is unavailable or presented very sporadically in the online process.

In this paper, we focus on the online semi-supervised learning problems (OS^{2}L). Several OS^{2}L algorithms have been proposed in the past several years. By using a heuristic method to greedily label the unlabeled examples, Babenko et al. [19] and Grabner et al. [20] tried to solve the OS^{2}L problems in an online supervised learning framework. Dyer et al. [21] presented a semi-supervised learning (SSL) framework called COMPOSE (COMPacted Object Sample Extraction), where a few labeled samples are given initially, and then a SSL problem is solved based on the currently labeled samples and new unlabeled samples, which follow a drift distribution. To reduce the computational complexity of manifold construction in the online training process, Kveton et al. [22] and Farajtabar et al. [23] proposed the harmonic solution for manifold regularization on an approximate graph.

By using online convex programming, Goldberg et al. [24] proposed an online manifold learning framework for SSL in a kernel space with stochastic gradient descent. In addition, they extended their method to online active learning by adding an optional component to select the instances to be labeled [25]. Sun et al. [26, 27] exploited the property of Fenchel conjugate of hinge loss and gradient ascend method to solve the dual problem of their online manifold learning model. Those algorithms in [24, 26, 27] are derived by using online gradient methods, implying that these methods can be regarded as solving the off-line semi-supervised learning models by stochastic gradient methods. However, none of these stochastic gradient methods can obtain an exact solution because they do not directly solve the constrained optimization problem involved.

In practice, we prefer an exact solution, which can usually achieve a more accurate result and meanwhile more efficiently. In this paper, we propose an algorithm with analytical solution to solve the online semi-supervised problem. Specifically, we propose a novel online manifold regularization learning model in a reproducing kernel Hilbert space (RKHS), by exploiting the internal geometry information of the unlabeled data and take advantage of the kernel methods. In each iteration of online training, by considering the new arrived sample and the previous samples, an online model based on a constrained optimization problem is presented, and the exact solution of the proposed model is obtained with the help of the Lagrange dual problem. Meanwhile, a fast learning algorithm (named FMOMR) is presented by introducing an approximate technique to compute the derivative of the manifold term. In addition, the regularization parameter of the proposed model can be regarded as a forgetting factor, which provides a reasonable and consistent way to control the number of support vectors. By such merits, the proposed online predictors experimentally exhibits a high accuracy comparable to batch algorithm LapSVM.

This paper substantially extend our previous work [28] by providing (a) a fast algorithm of the proposed model ("Fast Algorithm of the Proposed Model" section), (b) several buffering strategies ("Buffering Strategies" section), (c) a brief theory analysis of the proposed algorithms ("Theory Analysis" section), (d) more experiments ("Action Video Categorization" section), and (e) some background knowledge ("Background Knowledge" section).

The rest of this paper is organized as follows. The background knowledge is briefly reviewed in "Background Knowledge" section, and the proposed model and algorithms are detailed in "Online Manifold Learning with Kernels" section. After giving some experimental results in "Experiments" section, the paper is concluded by "Conclusion" section.

Background Knowledge

The background knowledge consists of two parts, Lagrange dual problem and LapSVM, a batch manifold learning model.

Lagrange Dual Problem

The Lagrange dual technique is frequently used for solving the primal optimization problem in RKHS. Thus, we give a brief review of the Lagrange dual problem in this section. Consider the primal constrained optimization problem:

$$\begin{array}{ll} \min_x & f_0(x) \\ \text{s.t.} & f_i(x) \le 0, \quad i = 1, ..., p \\ & h_i(x) = 0, \quad i = 1, ..., q, \end{array} \tag{1}$$

where $x \in R^n$ and $f_0(x)$ is an objective cost function that is minimized under the p inequality constraints $f_i(x) \le 0$ and q equality constraints $h_i(x) \le 0$. The Lagrange function of Eq. 1 is

$$L(x, \alpha, \beta) = f_0(x) + \sum_{i=1}^{p} \alpha_i f_i(x) + \sum_{i=1}^{q} \beta_i h_i(x) \tag{2}$$

where $\alpha = (\alpha_1, ..., \alpha_p)^T$ and $\beta = (\beta_1, ..., \beta_q)^T$ are Lagrange multipliers. By minimizing (2) over x, the Lagrange dual function g is defined as:

$$g(\alpha, \beta) = \min_x \quad L(x, \alpha, \beta) \tag{3}$$

Then the Lagrange dual problem of Eq. 1 is to maximize (3)

$$\begin{array}{ll} \max_{\alpha,\beta} & g(\alpha, \beta) \\ \text{s.t.} & \alpha_i \ge 0, \quad i = 1, ..., p. \end{array} \tag{4}$$

The strong duality, that is, the optimal value of Eqs. 1 and 4 being equal to each other, holds under the Slater's condition [29], that is, the primal problem is convex and there exists x_0 such that $f_i(x_0) < 0, i = 1, ..., p$. Therefore, the solution of the primal problem can be obtained by solving its Lagrange dual problem. Actually, the standard Laplacian

SVM (SVM based on manifold regularization) is commonly solved by Lagrange dual technique, as reviewed below.

Manifold Regularization for Semi-supervised Learning

Laplacian support vector machine [30] (LapSVM) is derived by adding the manifold regularization term into support vector machine (SVM). Given the labeled training data $(x_1, y_1), ..., (x_l, y_l)$ and unlabeled training data $x_{l+1}, ..., x_{l+u}$, where $x_i \in \mathcal{X}$ and $y_i \in \{-1, 1\}$, LapSVM is given by the following optimization problem:

$$\min_{f\in\mathcal{H}_K} \frac{1}{l}\sum_{i=1}^{l}(1-y_i f(x_i))_+ + \gamma_A||f||_K^2 + \frac{\gamma_I}{(u+l)^2}\mathbf{f}^T L\mathbf{f}. \tag{5}$$

where γ_A, γ_I are trade-off parameters, $\mathbf{f} = [f(x_1), ..., f(x_{l+u})]$, $K : \mathcal{X}\times\mathcal{X} \to \mathbb{R}$ is a Mercer kernel and $\mathcal{H}_K$ is an associated RKHS of functions $f : \mathcal{X} \to \mathbb{R}$ with the corresponding norm $\|\cdot\|_K$. Especially, the graph Laplacian L is defined by $L = D - W$, where W is the edge weights matrix and D is a diagonal matrix (defined by $D_{ii} = \sum_{j=1}^{l+u} W_{ij}$, $i = 1, ..., l+u$). By the Representer Theorem (see Theorem 2 in [30]), the optimal solution of Eq. 5 can be presented as

$$f^*(x) = \sum_{i=1}^{l+u}\alpha_i^* K(x, x_i). \tag{6}$$

By adding a bias term b to the above formula, the primal problem (5) can be rewritten as:

$$\begin{aligned}&\min_{\alpha\in\mathbb{R}^{l+u},\xi\in\mathbb{R}^l} \frac{1}{l}\sum_{i=1}^{l}\xi_i + \gamma_A\alpha^T K\alpha + \frac{\gamma_I}{(u+l)^2}\alpha^T KLK\alpha\\ &\text{s.t.}\quad y_i\left(\sum_{j=1}^{l+u}\alpha_j K(x_i, x_j) + b\right) \geq 1-\xi_i, i = 1, ..., l\\ &\xi_i \geq 0, i = 1, ..., l.\end{aligned}$$

where $\alpha = [\alpha_1, ..., \alpha_{l+u}]^T$. Then, we can obtain the Lagrangian:

$$\begin{aligned}&L(\alpha, \xi, b, \beta, \zeta)\\ &= \frac{1}{l}\sum_{i=1}^{l}\xi_i + \frac{1}{2}\alpha^T\left(2\gamma_A K + 2\frac{\gamma_A}{(l+u)^2}KLK\right)\alpha\\ &\quad - \sum_{i=1}^{l}\beta_i\left(y_i\left(\sum_{j=1}^{l+u}\alpha_j K(x_i, x_j) + b\right) - 1 + \xi_i\right)\\ &\quad - \sum_{i=1}^{l}\zeta_i\xi_i.\end{aligned}$$

where β_i, ζ_i are Lagrange multipliers.

To obtain the minimum with respect to b and ξ, consider the conditions $\partial L/\partial b = 0$ and $\partial L/\partial\xi_i = 0$. Thus, a reduced Lagrangian can be formulated as follows:

$$\begin{aligned}L^R(\alpha, \beta) &= \frac{1}{2}\alpha^T\left(2\gamma_A K + 2\frac{\gamma_I}{(u+l)^2}KLK\right)\alpha\\ &\quad -\alpha^T KJ^T Y\beta + \sum_{i=1}^{l}\beta_i.\end{aligned}$$

where $J = [I \quad 0]$ is an $l\times(l+u)$ matrix with I as the $l\times l$ identity matrix (assuming the first l points are labeled in the training set) and Y is a diagonal matrix with $Y_{ii} = y_i$ for $i = 1, ..., l$. By taking derivative of the reduced Lagrangian with respect to α, we have

$$\frac{\partial L^R}{\partial\alpha} = \left(2\gamma_A K + 2\frac{\gamma_I}{(u+l)^2}KLK\right)\alpha - KJ^T Y\beta. \tag{7}$$

So, we can get:

$$\alpha = \left(2\gamma_A I + 2\frac{\gamma_I}{(u+l)^2}LK\right)^{-1} J^T Y\beta^*. \tag{8}$$

Substituting back in the reduced Lagrangian, an optimization problem with respect to β is derived as follows:

$$\begin{aligned}\beta^* = &\max_{\beta\in\mathbb{R}^l}\sum_{i=1}^{l}\beta_i - \frac{1}{2}\beta^T Q\beta\\ &\text{s. t.}\quad \sum_{i=1}^{l}\beta_i y_i = 0\\ &\quad 0 \leq \beta_i \leq \frac{1}{l}, i = 1, ..., l\end{aligned} \tag{9}$$

where

$$Q = YJK\left(2\gamma_A I + 2\frac{\gamma_I}{(u+l)^2}LK\right)^{-1} J^T Y$$

By solving (9) and using the Eqs. 6 and 8, we can obtain the optimal solution $f^*(x)$. However, the process of training LapSVM classifier with all the training data can be very slow when the data size is large. To improve the computational efficiency for online learning, we proposed a novel online manifold learning model based on a constrained optimization problem, which is presented in the next section.

Online Manifold Learning with Kernels

In this section, the proposed model is presented in detail. In "Online Model Based on Manifold Regularization" section, a model based on manifold regularization is proposed for online semi-supervise learning in a RKHS. In "Online Algorithm of the Proposed Model" section, the proposed model is solved by exploiting the property of Lagrange dual problem. Several fast learning strategies are presented in "Fast Learning Strategies" section. A brief theoretical analysis of the proposed algorithms is presented in "Theory Analysis" section.

Online Model Based on Manifold Regularization

Assume that the current learning data for semi-supervised learning are $(x_1, y_1, \delta_1), (x_2, y_2, \delta_2), ..., (x_t, y_t, \delta_t)$ where $x_i \in \mathcal{X}$ is a point, $y_i \in \mathcal{Y} = \{-1, 1\}$ is its label and δ_i is a flag to determine whether the label y_i is available (y_i is available if and only if $\delta_i = 1$). At round t, the current predictor is $h_t(x) = \text{sign}(f_t(x))$ and f_0 is set as $f_0 = 0$ in our algorithm. In online semi-supervised learning, when a new sample $(x_{t+1}, y_{t+1}, \delta_{t+1})$ is available, the function f_{t+1} is updated based on the current decision function f_t and the implicit feedback, that is, the manifold structure of the samples. The detailed process of online manifold learning is

presented in Fig. 1. In Fig. 1, a new input x_t is provided to the current predictor and the decision value $f(x_t)$ is computed by the predictor. Thereafter, the learner will update the decision function in different ways based on different feedbacks: if the label of x_t is available, the learner will update the classifier with both the explicit feedback y_t and the implicit feedback under the manifold structure of the samples; otherwise, the classifier will be updated only based on the implicit feedback. The process will continue until no more new samples arrive and the final predictor $h_T(x) = \text{sign}(f_T(x))$ (T is the final time) is derived for classification tasks.

Suppose that $K(\cdot,\cdot)$ is a chosen Kernel function over the training samples and $\mathcal{H}$ is the corresponding RKHS. Therefore, according to the Representer Theory [31], we can write f_t and f_{t+1} as follows:

$$\begin{aligned} f_t(\cdot) &= \sum_{i=1}^{t} \alpha_i^t K(x_i,\cdot), \\ f_{t+1}(\cdot) &= \sum_{i=1}^{t} \alpha_i^{t+1} K(x_i,\cdot) + \alpha_{t+1}^{t+1} K(x_{t+1},\cdot). \end{aligned} \tag{10}$$

In the online learning process, our aim is to update $\{\alpha_i^{t+1}\}_{i=1}^{t+1}$ from $\{\alpha_i^t\}_{i=1}^{t}$ based on a proper algorithm. Considering the trade-off between the amount of progress made on each round and the amount of information retained from previous rounds, and compromise the classification error, the manifold constraint and the complexity of f as LapSVM, our online semi-supervised learning model with manifold regularization is presented as follows:

$$\begin{aligned} \min_{f,\xi_{t+1}} \ & \tfrac{1}{2}\|f - f_t\|_{\mathcal{H}}^2 + \tfrac{\lambda_1}{2}\|f\|_{\mathcal{H}}^2 + C\delta_{t+1}\xi_{t+1} \\ & + \tfrac{1}{2}\lambda_2 \sum_{i=1}^{t} (f(x_i) - f(x_{t+1}))^2 w_{it+1} \\ \text{s.t.} \ & y_{t+1} f(x_{t+1}) \geq 1 - \xi_{t+1}, \xi_{t+1} \geq 0 \end{aligned} \tag{11}$$

where $\frac{1}{2}\|f - f_t\|_{\mathcal{H}}^2$ measures the difference between f and the previous f_t, $\|f\|_{\mathcal{H}}^2$ controls the complexity of the decision function f, $\sum_{i=1}^{t}(f(x_i) - f(x_{t+1}))^2 w_{it+1}$ is the manifold regularizer which depends on the edge weight w_{it+1}, f and x_i, and ξ_{t+1} is the slack variable denoting a possible error for the newly arrived data $(x_{t+1}, y_{t+1}, \delta_{t+1})$ after f is determined, λ_1, λ_2 and C are parameters reflecting the weights compromising complexity, the manifold regularizer and the classification error.

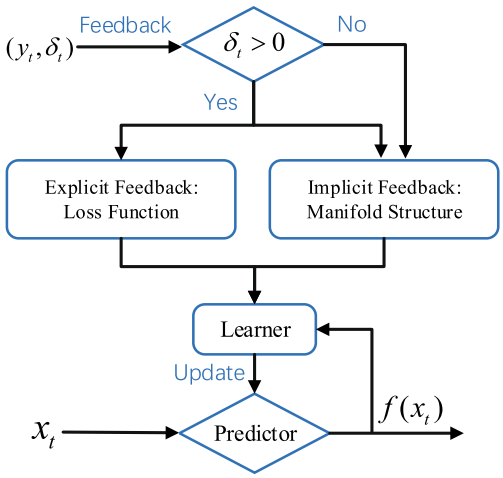

Fig. 1 Online semi-supervised learning based on manifold regularization framework

In the objective function of Eq. 11, the manifold structure of the samples is reflected in the term $\sum_{i=1}^{t}(f(x_i) - f(x_{t+1}))^2 w_{it+1}$, which can be regarded as an implicit feedback. This regularization term makes the new sample gain a similar decision value to its close sample in the manifold. The solution of the proposed model is presented in the next section.

Online Algorithm of the Proposed Model

In this section, we give a detailed solution of the proposed model by exploiting the property of Lagrange dual problem. Assuming that $\delta_{t+1} = 1$ (if $\delta_{t+1} = 0$, the solution of Eq. 11 can be obtained by the similar process as bellow), the Lagrange dual problem of Eq. 11 is

$$\begin{aligned} \max_{\gamma_{t+1}} \min_{f,\xi_{t+1}} \ & L(f, \xi_{t+1}, \gamma_{t+1}, \beta_{t+1}) \\ \text{s.t.} \ & \gamma_{t+1} \geq 0, \quad \beta_{t+1} \geq 0 \end{aligned} \tag{12}$$

Algorithm 1 Model based online manifold regularization (MOMR) algorithm

Input: Parameters:$\lambda_1 \geq 0$(default$= 10^{-3}$),$\lambda_2 \geq 0$(default$= 10^{-3}$),$C \geq 0$(default$= 1$)

Initialize: $f = 0$.
Receive an incoming instance:x_1;
Let $\alpha^1 = 1$, $f = \alpha^1 K(x_t,\cdot)$
for $t = 2; i \leq T; t++$ **do**
  Receive an incoming instance: x_t;
  Receive the flag δ_t and the label y_t;
  Update the Gram Matrix K with x_t;
  Let $\tilde{\alpha}^t = [\alpha^{t-1}; 0]$;
  Compute D and W by (23) and (24).
  Let $L = D - W$;
  Compute $A = K + \lambda_1 K + \lambda_2 KLK$;
  if $\delta_t == 0$ **then**
    $\alpha^t = A^{-1} K \tilde{\alpha}^t$;
  else
    Compute $J = Ke$, where $e = [0, ..., 0, 1]^T$ is a t-dimensional vector;
    Compute γ_t^* by (23);
    $\alpha^t = A^{-1}(K\tilde{\alpha}^t + y_t \gamma_t^* J)$;
  end if
  $f = \sum_i \alpha_i^t K(x_i,\cdot)$,
  where α_i^t is the i-th element of α^t;
end for

Output: f.

where γ_{t+1} and β_{t+1} are the Lagrange multipliers corresponding to the constraints $y_{t+1}f(x_{t+1}) \geq 1-\xi_{t+1}$ and $\xi_{t+1} \geq 0$, respectively, and

$$\begin{aligned} L(f,\xi_{t+1},\gamma_{t+1},\beta_{t+1}) = & \tfrac{1}{2}\|f-f_t\|^2_{\mathcal{H}} + \tfrac{\lambda_1}{2}\|f\|^2_{\mathcal{H}} \\ & + \tfrac{1}{2}\lambda_2 \sum_{i=1}^{t}(f(x_i)-f(x_{t+1}))^2 w_{it+1} \\ & - \gamma_{t+1}(y_{t+1}f(x_{t+1}) - 1 + \xi_{t+1}) \\ & + C\xi_{t+1} - \beta_{t+1}\xi_{t+1} \end{aligned}$$

By solving the Lagrange dual problem of Eq. 11 (the details can be found in the Appendix), we can obtain the new classifier at time $t+1$:

$$\begin{aligned} f_{t+1}(x) &= \sum_{i=1}^{t+1} \alpha_i^{t+1} K(x_{t+1}, x), \\ h_{t+1} &= \text{sign}(f_{t+1}(x)), \end{aligned} \tag{13}$$

where

$$\alpha^{t+1} = A^{-1}(K\tilde{\alpha}^t + \delta_{t+1}y_{t+1}\gamma^*_{t+1}J).$$

The above process is summarized in Algorithm 1. In Algorithm 1, when the first sample arrives, the value of α_1 is set to be 1.

However, there are two difficulties in performing Algorithm 1: (1) To compute the value of α_t, we need to compute the inverse of matrix $A = K + \lambda_1 K + \lambda_2 KLK$, which is difficult to calculate when t is very large; (2) In the online learning process, the online manifold learning algorithms with kernel functions have to store the sequence up to the current round. In a result, the set of support vectors will grow unboundedly, which limits the applicability of the online algorithms. Therefore, we present a fast algorithm to solve the proposed model and introduce several buffering strategies to reduce the number of support vectors.

Fast Learning Strategies

Fast Algorithm of the Proposed Model

In this section, we propose a fast algorithm to solve the proposed model. Note that if we let $\lambda_2 = 0$, the process of calculating the inverse matrix can be avoided. There, for the sake of taking advantage of the properties of manifold regularization and improving the computational efficiency, we use an approximate term to replace (31).

Consider the formula (30), by replacing the term $\lambda_2 KLK\alpha$ with $\lambda_2 KLK\alpha_t$, we have

$$\begin{aligned} \frac{\partial L^R}{\partial \alpha} \approx & (K + \lambda_1 K)\alpha + \lambda_2 KLK\alpha_t \\ & - K\alpha^t - Jy_{t+1}\gamma_{t+1} \end{aligned} \tag{14}$$

Algorithm 2 Fast algorithm to model based online manifold regularization (FMOMR)

Input: Parameters:$\lambda_1 \geq 0$(default= 10^{-3}),$\lambda_2 \geq 0$(default= 10^{-3}),$C \geq 0$(default= 1)

```
Initialize: f = 0.
Receive an incoming instance:x_1;
Let α^1 = 1, f = α^1 K(x_t, ·)
for t = 2; i ≤ T; t++ do
    Receive an incoming instance:x_t;
    Receive the flag δ_t and the label y_t;
    Update the Gram Matrix K with x_t;
    Let α̃^t = [α^{t-1}; 0];
    Compute D and W by (23) and (24).
    Let L = D − W;
    if δ_t == 0 then
        α = 1/(1+λ_1) (I − λ_2 LK)α^t;
    else
        Compute J = Ke, where e = [0, ..., 0, 1] is a
        t-dimensional vector;
        Compute γ_t by (17);
        α = 1/(1+λ_1) [(I − λ_2 LK)α^{t-1} + e y_t γ_t];
    end if
    f = Σ_i^t α_i^t K(x_i, ·),
    where α_i^t is the i-th element of α^t;
end for
```

Output: f.

This approximation is reasonable for that the term $\frac{1}{2}\|f-f_t\|^2_{\mathcal{H}}$ is used to control the distance of a predicted f from the previous f_t in our model, which can guarantee that the difference between α^{t+1} and α^t is not very large. In addition, the convex function $M(\alpha) = \alpha^T \lambda_2 KLK\alpha$ is continuous and differentiable, so we have

$$\frac{\partial M}{\partial \alpha^{t+1}} \approx \frac{\partial M}{\partial \alpha^t}$$

that is,

$$\lambda_2 KLK\alpha^{t+1} \approx \lambda_2 KLK\alpha^t.$$

Now, from Eq. 14, we get

$$\alpha = \tfrac{1}{1+\lambda_1}[(I - \lambda_2 LK)\alpha_t + ey_{t+1}\gamma_{t+1}] \tag{15}$$

Taking the derivative of Eq. 29 with respect to γ_{t+1} we get:

$$\frac{\partial L^R}{\partial \gamma_{t+1}} = 1 - y_{t+1}\alpha^T J = 0 \tag{16}$$

Substituting (15) into (16), we have

$$\overline{\gamma}_{t+1} = 1 + \lambda_1 - y_{t+1}J^T(I - \lambda_2 LK)\alpha^t \tag{17}$$

Let the approximate solution of Eq. 30 be $\hat{\alpha}_{t+1}$ and $\hat{\gamma}^*_{t+1}$. Hence,

$$\hat{\gamma}^*_{t+1} = \begin{cases} 0, & \text{if } \overline{\gamma}_{t+1} \leq 0 \\ C, & \text{if } \overline{\gamma}_{t+1} \geq 0 \\ \overline{\gamma}_{t+1}, & \text{otherwise} \end{cases} \tag{18}$$

Similar to Eq. 13, the classifier obtained at time $t+1$ is:

$$\begin{aligned} f_{t+1}(x) &= \sum_{i=1}^{t+1} \hat{\alpha}_i^{t+1} K(x_{t+1}, x), \\ h_{t+1} &= \text{sign}(f_{t+1}(x)), \end{aligned} \tag{19}$$

where

$$\hat{\alpha}_{t+1} = \frac{1}{1+\lambda_1}\left[(I - \lambda_2 LK)\alpha_t + e\delta_{t+1} y_{t+1} \hat{\gamma}^*_{t+1}\right]$$

The above process is summarized in Algorithm 2.

The main computation in Eqs. 15 and 17 is to calculate the matrix multiplication $L \times K$. It can be seen that $L = D - W$ is a sparse matrix by its definition (D is a diagonal matrix and W is a matrix that only $2t$ elements are non-zero), which means the computational complexity of $L \times K$ is only $O(t^2)$. Therefore, the computational complexity is $O(t^2)$ of Algorithm 2. Note that the computational complexity becomes very high with the increasing of t. This limits the scalability of the proposed algorithms. Therefore, we present several buffering strategies to improve the scalability of the online algorithms in the next section.

Buffering Strategies

In practice, kernel-based discriminative algorithms have been shown to perform very well on semi-supervised learning problems [30, 32]. However, in the online learning process, the set of support vectors will grow unboundedly, which limits the applicability of the online manifold regularization algorithms. To address this problem, we present several approaches to bound the size of the support set.

Buffering strategies [5, 24, 33] keep a fixed number of support vectors for online learning. Let the buffer size be τ. There are several different strategies:

(1) Buffer-N. The oldest sample in the buffer is replaced with the new incoming sample after each online learning round.
(2) Buffer-U. The oldest unlabeled sample in the buffer is replaced with the new incoming sample after each online learning round. When the buffer is filled with labeled samples, the oldest labeled points is evicted from the buffer.

To modify this for a more general case, we remove the sample with the smallest $|\alpha_i^t|$ in round t, where $|\cdot|$ is the absolute value symbol. As suggested in [24], we choose Buffer-U as the buffering strategy for all our experiments.

Theory Analysis

In this section, we give out a brief theory analysis of the proposed algorithms.

Theorem 1 *Suppose that $K(\cdot,\cdot)$ is a chosen Kernel function over the training samples and $\mathcal{H}$ is the corresponding RKHS, then* (13) *is exactly the solution of the primal problem* (11).

Proof Let

$$\begin{aligned} c_t^1(f) &= 1 - \xi_{t+1} - y_{t+1} f(x_{t+1}), \\ c_t^2(f) &= -\xi_{t+1}. \end{aligned}$$

Apparently, $c_t^1(f)$ and $c_t^2(f)$ are continuous. In addition, since the object function of Eq. 11 is convex, by the Convex Duality Theorem of [34] (see the Theorem 14.37 on page 532), the optimal value of the primal problem (11) is equal to that of its Lagrange dual problem. Therefore, according to the above derivational process, the result (13) is exactly the solution of the primal problem (11). □

Theorem 1 implies that MOMR is an exact algorithm with respect to the proposed model (11). Next, we give out an analysis of the relationship between the proposed MOMR and FMOMR.

Theorem 2 *Suppose $\lambda_2 = 0$, then the solution of* Eqs. 13 *and* 19 *is equivalent.*

Proof Suppose $\lambda_2 = 0$, we have

$$\begin{aligned} J^T A^{-1} J &= e^T K (K + \lambda_1 K)^{-1} K e \\ &= \frac{1}{1+\lambda_1} e^T K e = \frac{1}{1+\lambda_1}. \end{aligned} \tag{20}$$

Therefore, from Eq. 33, we get

$$\begin{aligned} \overline{\gamma}_{t+1} &= \frac{1 - y_{t+1} J^T A^{-1} K \tilde{\alpha}^t}{J^T A^{-1} J} \\ &= 1 + \lambda_1 - y_{t+1} J^T \tilde{\alpha}^t. \end{aligned} \tag{21}$$

which is equivalent to Eq. 17 if $\lambda_2 = 0$.

Similarly, by substituting $\lambda_2 = 0$ into (31) and (15) respectively, we have

$$\hat{\alpha}_{t+1} = \alpha_{t+1} = \frac{1}{1+\lambda_1}(\tilde{\alpha}^t + e y_{t+1} \gamma_{t+1}). \tag{22}$$

□

Theorem 2 is reasonable for that the Algorithm 2 is obtained only by approximating the derivative of the manifold regularization term. In addition, for that (31) and (15) are continuous with respect to λ_2, (15) is an appropriate approximation of Eq. 31 when λ_2 is very small.

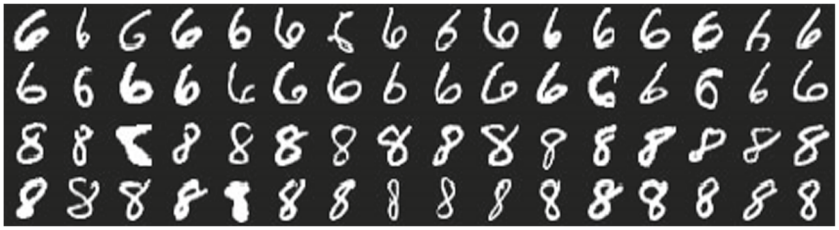

Fig. 2 Some images of the MNIST3VS6. The *top two rows* are images of "6" while the *bottom two rows* are images of "8"

Experiments

In this section, to verify the effectiveness, we compare the proposed algorithms, MOMR and FMOMR, with two online manifold regularization algorithms and a batch algorithm on three data sets (see "Handwritten Digit Recognition—4" section), respectively.

In all the experiments, the RBF kernel $k(x_i, x_j) = \exp(-\|x_i - x_j\|^2/(2\sigma_K^2))$ is used for classification. The edge weight is $k(x_i, x_j) = \exp(-\|x_i - x_j\|^2/(2\sigma_W^2))$, which define a fully connected graph. The labeled rate of training samples is 2%.

In our experiments, we focus on online manifold regularization algorithms derived from the dual problem. Therefore, we compare the performance of our algorithms with an online manifold regularization algorithm based on Example-Associate Update (denoted by OMR-EA), an online manifold regularization algorithm based on Overall Update (denoted by OMR-Overall) [26], and a batch manifold regularization algorithm LapSVM [30]. As suggested in [26], the step sizes of the OMR-EA and OMR-Overall are set to be a small value 0.01.

All the evaluations share the same buffering strategy, Buffer-U, but employ different buffer sizes ($B \in \{50, 100, 150, 200\}$). The parameter values $\sigma_K, \sigma_W, \lambda_1$ and λ_2 are selected by using five-fold cross validation on the first 500 samples of the training data, where $\sigma_K, \sigma_W \in \{2^{-3}, 2^{-2}, 2^{-1}, 2^{-0}, 2^1, 2^2, 2^3\}$ and $\lambda_1, \lambda_2 \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{-0}, 10^1, 10^2\}$. In addition, the value of parameter C is set to be 1 for the proposed algorithm MOMR and FMOMR. The computational efficiencies of all the algorithms are evaluated in terms of their CPU running time (in seconds). All the experiments are implemented in Matlab on a PC with Inter(R) Core(TM) 3.2 GHz CPU, 4G RAM and Windows 7 operating system.

All the four online algorithms are performed in the same way which can be divided into two steps: (1) Online processing: training a classifier with a new arrived sample using an online algorithm. (2) Test: testing the final model on a test set. However, the batch algorithm LapSVM is trained with all the visible samples in each learning round. We repeat all the experiments ten times (each with an independent random permutation of the training samples) and the results presented bellow are all average over ten trials.

Handwritten Digit Recognition

In this section, we perform an evaluation experiment on the MNIST data set [35]. We focus on the binary classification task of separating "6" from "8" (MNIST6VS8) in our experiment. The sizes of the training set and test set are 11769 and 1932 respectively. Some images of the MNIST6VS8 data set are presented in Fig. 2.

The test accuracies are summarized in Table 1. From the results, the test accuracies of MOMR and FMOMR are comparable to those of LapSVM and higher than those of OMR-EA and OMR-Overall. This is reasonable for that MOMR is exactly the solution of the proposed model, while OMR-EA and OMR-Overall [26] are obtained by using the stochastic gradient methods. And, the performance of the fast algorithm FMOMR is very similar to that of the algorithm MOMR. It implies that FMOMR is a proper approximate solution to the proposed model.

The online updating time is presented in Fig. 3. We can see that: with respect to the updating time (a) MOMR is comparable to the other three online algorithms when the

Table 1 On the MNIST6VS8, test accuracies (%) of MOMR, FMOMR, OMR-EA, OMR-Overall, and LapSVM with using different buffer sizes

B	MOMR	FMOMR	OMR-EA	OMR-Overall	LapSVM
50	*98.012 ± 0.442*	97.940±0.493	96.491±1.775	97.495±0.714	98.030±0
100	98.613±0.318	*98.685±0.206*	98.427±0.265	97.940±0.553	98.030±0
150	*99.068±0.146*	*99.068±0.073*	98.913±0.073	97.904±0.622	98.030±0
200	99.048±0.078	*99.120±0.097*	98.954±0.177	97.981±0.543	98.030±0

Note that LapSVM is an offline algorithm for manifold regularization, and is independent of the buffer size. The best classification results with each buffer size are marked in italics

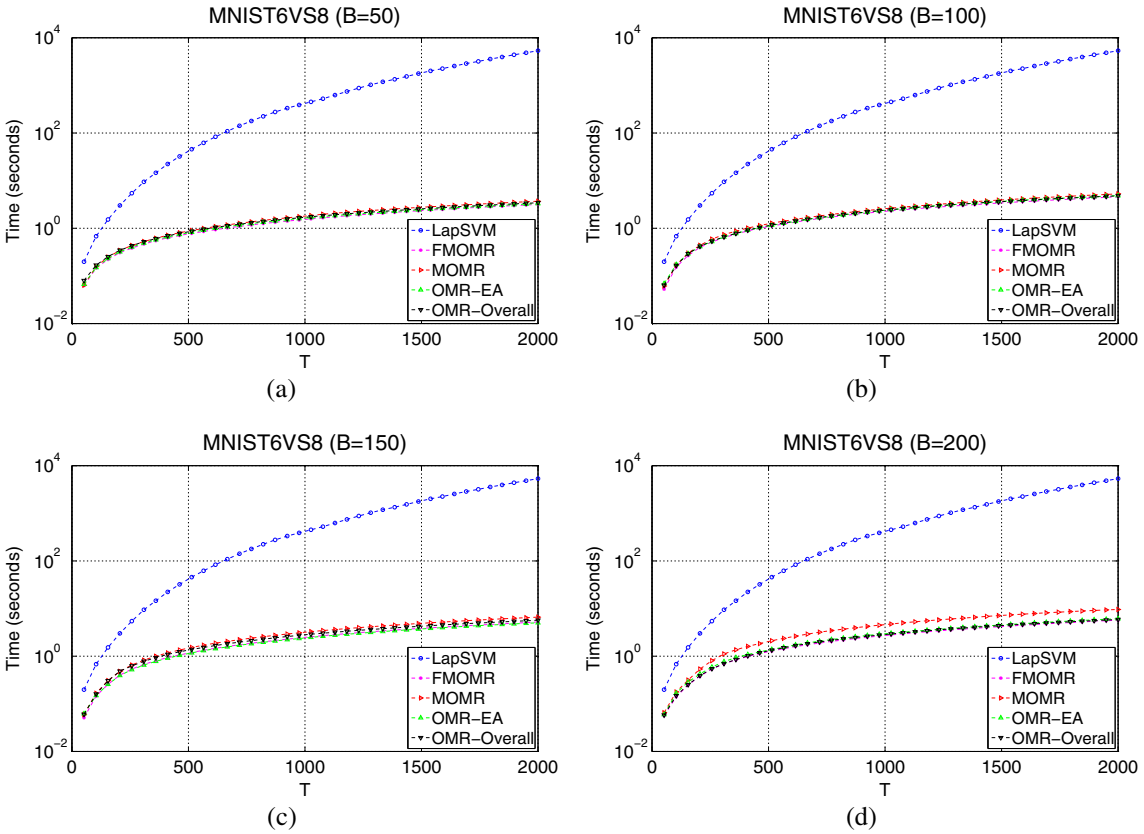

Fig. 3 Cumulative running time of online updating the classifiers with different buffer sizes on the MNIST6VS8 data set

buffer size is small; (b) FMOMR is comparable to the online algorithms OMR-EA and OMR-Overall, and much faster than the off-line algorithm LapSVM. These are reasonable for that each sample is trained only once by the online algorithms and a buffering strategy is used to reduce the repeated training process.

Face Recognition

This experiment is performed on the data set FACEMIT [36] which contains 361-dimensional images of faces and non-faces. A balanced subset (size 5000) from FACEMIT

Fig. 4 Some images of the FACEMIT. The *top four rows* are images of faces while the *bottom four rows* are images of non-faces

Table 2 On the FACEMIT, test accuracies (%) of MOMR, FMOMR, OMR-EA, OMR-Overall, and LapSVM with different buffer sizes

B	MOMR	FMOMR	OMR-EA	OMR-Overall	LapSVM
50	*78.024 ± 3.411*	*78.024 ±3.411*	77.992±3.390	78.000±3.478	77.600±0
100	78.412±3.270	*78.420 ± 3.316*	78.080±3.142	78.048±3.253	77.600±0
150	78.528±3.332	*78.560 ± 3.347*	77.996±3.129	77.960±3.240	77.600±0
200	78.552±3.360	*78.608 ± 3.363*	77.948±3.126	77.920±3.237	77.600±0

The best classification results with each buffer size are marked in italics

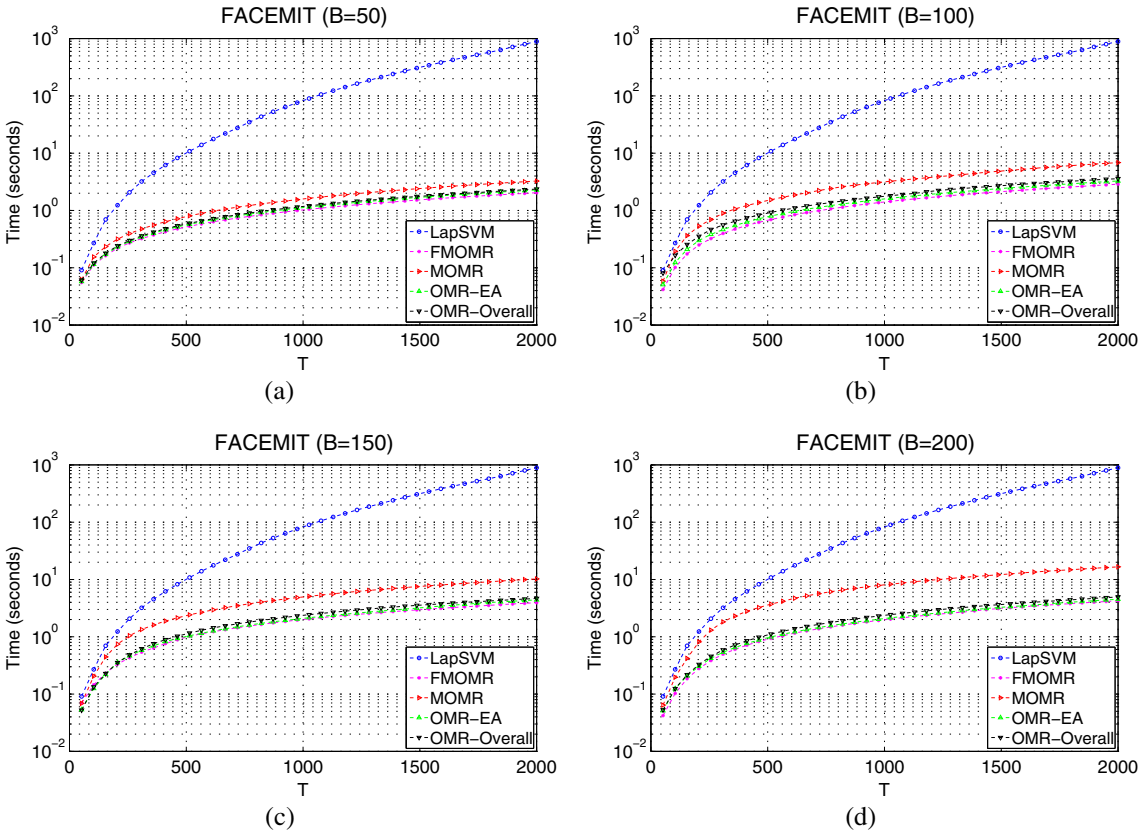

Fig. 5 Cumulative running time of online updating the classifiers with different buffer sizes on the FACEMIT data set

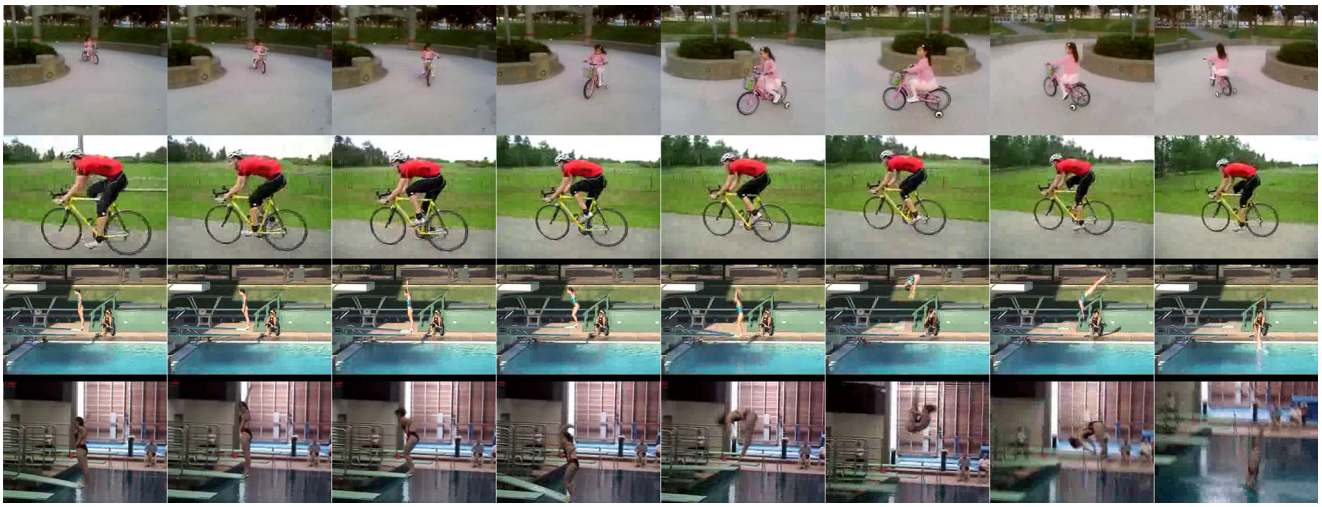

Fig. 6 Some frames from the videos of biking and diving. The *top two rows* are frames of biking while the *bottom two rows* are frames of diving

Table 3 Test accuracies (%) of MOMR, FMOMR, OMR-EA, OMR-Overall, and LapSVM on a subset of UCF YouTube [29] with different buffer sizes

Buffer	MOMR	FMOMR	OMR-EA	OMR-Overall	LapSVM
50	85.260±2.151	*85.272±2.142*	83.094±2.3475	82.994±1.930	95.05±0
100	91.374±1.729	*91.377±1.727*	89.344±1.435	89.349±1.323	95.05±0
150	*93.480±1.375*	93.473±1.335	93.440±1.089	93.443±1.171	95.05±0
200	*94.988±0.158*	*94.988±0.158*	94.830±0.282	94.841±0.310	95.05±0

The best classification results with each buffer size are marked in italics

is randomly sampled and divided into two sets obeying a rule that the number of training samples is equal to that of test samples. Some images of the FACEMIT data set are presented in Fig. 4.

The test accuracies are summarized in Table 2. We can make the following comments: (a) The test accuracies of MOMR and FMOMR are higher than those of the other algorithms; (b) FMOMR surpass other algorithms with respect to the test accuracy, which further demonstrates that the proposed fast approximate algorithm FMOMR is reasonable and efficient.

The online updating time of the five algorithms are presented in Fig. 5. It can be seen that FMOMR is the fastest algorithm among all the five algorithms. Additionally, the difference between MOMR and FMOMR increases with the increasing of buffer size. This can be explained by that the computational complexity of MOMR and FMOMR are $O(B^3)$ and $O(B^2)$ respectively. Note that in Fig. 5, the curves are plotted by using singe logarithmic coordinate axis. Therefore, MOMR consumes more time than FMOMR as B increases. However, when B is small, both MOMR and FMOMR can be implemented very fast, so that the difference of cumulative running time between MOMR and FMOMR is insignificant.

Action Video Categorization

Further, we evaluate our methods on a kind of multi-manifold data, action video. As we know, a video is always made up of lots of static images which keep coherence in content and space, especially action videos. We adopt the UCF YouTube dataset [37] which consists of 1168 video

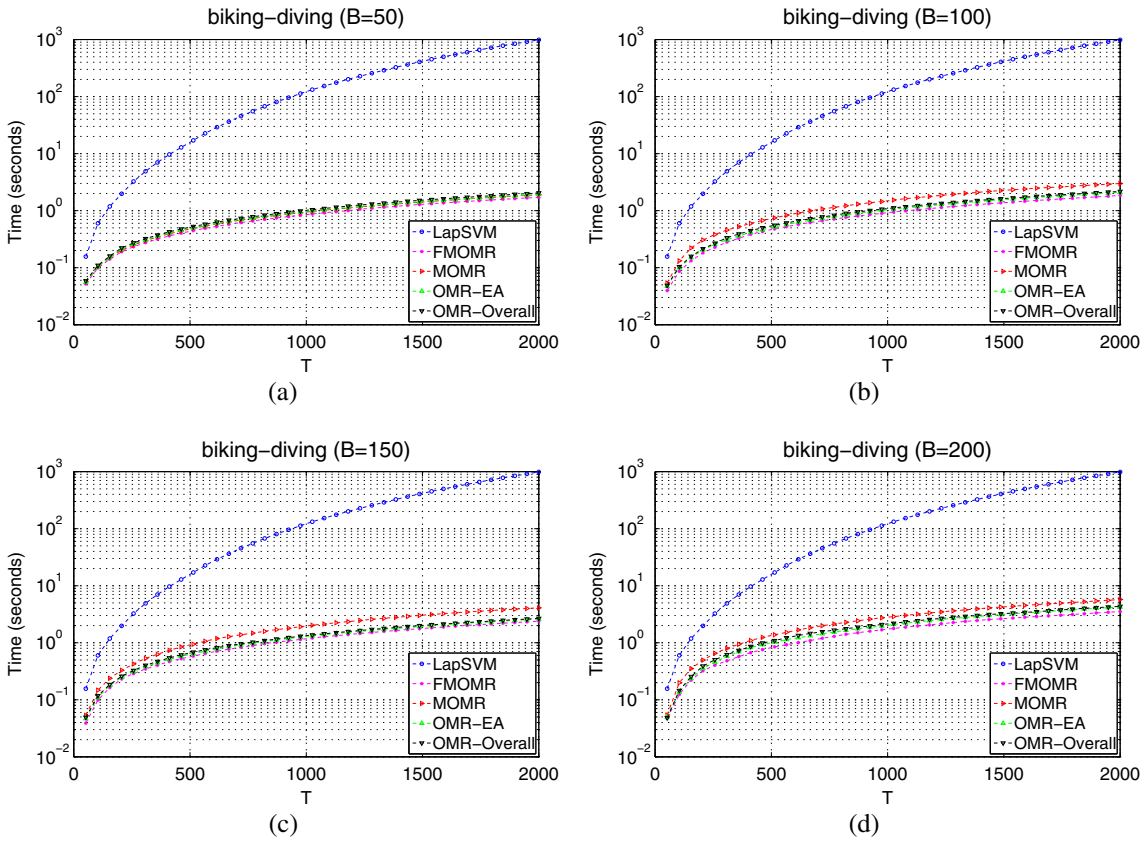

Fig. 7 Cumulative running time of online updating the classifiers with different buffer sizes on the UCF YouTube dataset

sequences captured under uncontrolled conditions. It is a challenging dataset owing to tremendous variations in camera motion, object pose, cluttered background, viewpoint, illumination, etc. We select two action categories, biking and diving (some images are presented in Fig. 6.), which both have a better continuity. We utilize the dense trajectories [38] to describe actions in the videos. The method can extract essential features representing actions which is robust to fast irregular motions and short boundaries. Then, 10,000 frames are sampled from these 2 action categories respectively (so the total number of frames is 20,000). They are divided into two sets: the training set and the test set with a proportion 1:1 for our experiment. The task is to classify these frames into these two action categories.

The test accuracies are summarized in Table 3. We can make the following comments: (a) The test accuracies of MOMR and FMOMR are comparable with the off-line algorithm LapSVM and higher than those of the two online algorithms OMR-EA and OMR-Overall; (b) the proposed algorithms MOMR and FMOMR make 2% improvements over the other two online algorithms when the buffer size is small; and (c) all the online algorithms make a significant improvement on performance with the increasing of the buffer size.

The online updating time of the five algorithms are presented in Fig. 7. With respect to the running time, it can be seen that (a) FMOMR is faster than the other compared online algorithms, and (b) all the compared online algorithms are much faster than the off-line algorithm LapSVM.

Considering the above results, it can be inferred that the proposed algorithms can reach the first grade among the five algorithms both on the test accuracy aspect and on the running time aspect. The proposed fast algorithm FMOMR is the best among the compared online algorithms for its performances on the three data sets because (a) FMOMR is the fastest algorithm among the compared algorithm; (b) in the aspect of generalization performance, FMOMR is better than OMR-EA and OMR-Overall and FMOMR has a comparable performance to the batch algorithm LapSVM.

Additionally, the test accuracy is higher with a larger buffer, but the time cost increases with the increase of the buffer size. In practice, the buffer size can be used to trade-off the accuracy and the time cost of online classifiers. An appropriate buffer size can be derived by using cross validation on the first N arrived samples, where N is a predefined number.

Conclusion

According to the manifold regularized online model, we give out an analytical solution of the constrained optimization problem by exploiting the techniques of the Lagrange dual problem. The proposed idea offers two new algorithms to solve the online semi-supervised leaning problem. Experiment results verify the effectiveness and validity of the proposed algorithms.

In fact, the proposed algorithms can solve not only semi-supervised learning problems but also online supervise learning problems (this can be done in the algorithm MOMR by deleting the manifold regularization term from the object function of (11) and setting the value of λ_2 to be 0). In the future work, we will extend the proposed algorithms to solve some specific online learning problems.

Acknowledgment This work is partly supported by NSFC grants 61375005, U1613213, 61210009, 61627808, 61603389, 61602483, MOST grants 2015BAK35B00, 2015BAK35B01, Guangdong Science and Technology Department grant 2016B090910001, and BNSF grant 4174107.

Compliance with Ethical Standards

Conflict of interests We declare that we have no conflict of interest.

Human and Animal Rights This article does not contain any studies with human participants or animals performed by any of the authors.

Appendix

In this Appendix, we give out the derivation process of Eq. 13.

For simplicity, we define D and W as

$$D_{ij} = \begin{cases} w_{ij} & \text{if} \quad 0 < i = j < t+1 \\ \sum_{i=1}^{t} w_{it+1} & \text{if} \quad i = j = t+1 \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

$$W_{ij} = \begin{cases} w_{ij} & \text{if} \quad 0 < i < t+1, j = t+1 \\ w_{ij} & \text{if} \quad i = t+1, 0 < j < t+1 \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

Substituting (10), (23), (24) into (12) and letting $L = D - W$, we have

$$\begin{aligned} L(\alpha, \xi_{t+1}, \gamma_{t+1}, \beta_{t+1}) = & \tfrac{1}{2}\alpha^T(K + \lambda_1 K + \lambda_2 KLK)\alpha \\ & -\gamma_{t+1}(y_{t+1}\alpha^T J - 1 + \xi_{t+1}) \\ & -\alpha^T K\tilde{\alpha}^t - \beta_{t+1}\xi_{t+1} + C\xi_{t+1} + c_0 \end{aligned} \tag{25}$$

where $\alpha = [\alpha_1, ..., \alpha_{t+1}]^T$, $\tilde{\alpha}^t = [\alpha_1^t, ..., \alpha_t^t, 0]^T$, K is a $(t+1)\times(t+1)$ Gram Matrix with $K_{ij} = K(x_i, x_j)$, $J = Ke$, $e = [0, ..., 0, 1]^T$ is a $(t+1)$-dimensional vector and c_0 is a constant.

Note that $L(\alpha, \xi_{t+1}, \gamma_{t+1}, \beta_{t+1})$ attains its minimum with respect to α and ξ_{t+1}, if and only if the following conditions are satisfied:

$$\nabla_\alpha L(\alpha, \xi_{t+1}, \gamma_{t+1}, \beta_{t+1}) = 0, \tag{26}$$

$$\nabla_{\xi_{t+1}} L(\alpha, \xi_{t+1}, \gamma_{t+1}, \beta_{t+1}) = 0. \tag{27}$$

Therefore, we have

$$\begin{aligned} \frac{\partial L}{\partial \xi_{t+1}} &= -\gamma_{t+1} - \beta_{t+1} + C = 0 \\ \Longrightarrow \quad & 0 \le \gamma_{t+1} \le C. \end{aligned} \tag{28}$$

According to the above identity, we formulate a reduced Lagrangian:

$$\begin{aligned} L^R(\alpha, \gamma_{t+1}) = & \tfrac{1}{2}\alpha^T(K + \lambda_1 K + \lambda_2 KLK)\alpha \\ & -\gamma_{t+1}(y_{t+1}\alpha^T J - 1) \\ & -\alpha^T K\tilde{\alpha}^t + c_0. \end{aligned} \tag{29}$$

Taking derivative of Eq. 29 with respect to α, we have:

$$\begin{aligned} \frac{\partial L^R}{\partial \alpha} = & (K + \lambda_1 K + \lambda_2 KLK)\alpha \\ & -K\tilde{\alpha}^t - J y_{t+1}\gamma_{t+1}. \end{aligned} \tag{30}$$

Note that $\partial L^R/\partial\alpha = 0$. Therefore, we have:

$$\begin{aligned} \alpha = & (K + \lambda_1 K + \lambda_2 KLK)^{-1} \times \\ & (K\tilde{\alpha}^t + J y_{t+1}\gamma_{t+1}). \end{aligned} \tag{31}$$

Substituting (31) back into the reduced Lagrangian (29), we get:

$$\begin{aligned} \max_{\gamma_{t+1}} \; & -\tfrac{1}{2}(K\tilde{\alpha}^t + J y_{t+1}\gamma_{t+1})^T A^{-1}(K\tilde{\alpha}^t + J y_{t+1}\gamma_{t+1}) \\ & +\gamma_{t+1} \\ \text{s.t.} \quad & 0 \le \gamma_{t+1} \le C, \end{aligned} \tag{32}$$

where $A = K + \lambda_1 K + \lambda_2 KLK$.

Let $\overline{\gamma}_{t+1}$ be the stationary point of the object function of Eq. 32.

Therefore,

$$\overline{\gamma}_{t+1} = \frac{1 - y_{t+1} J^T A^{-1} K\tilde{\alpha}^t}{J^T A^{-1} J}. \tag{33}$$

Assume that the optimal solution of Eq. 32 is γ^*_{t+1} . Note that the object function (32) is quadratic, so the optimal solution γ^*_{t+1} in the interval $[0, C]$ is at either 0, C or $\overline{\gamma}_{t+1}$. Hence

$$\gamma^*_{t+1} = \begin{cases} 0, & \text{if} \quad \overline{\gamma}_{t+1} \le 0 \\ C, & \text{if} \quad \overline{\gamma}_{t+1} \ge 0 \\ \overline{\gamma}_{t+1}, & \text{otherwise} \end{cases} \tag{34}$$

Furthermore, if $\delta_{t+1} = 0$, we can obtain the solution of the proposed model by the similar process as above. Thus, the classifier obtained at time $t + 1$ is:

$$\begin{aligned} & f_{t+1}(x) = \sum_{i=1}^{t+1} \alpha_i^{t+1} K(x_{t+1}, x), \\ & h_{t+1} = \text{sign}(f_{t+1}(x)), \end{aligned} \tag{35}$$

where

$$\alpha^{t+1} = A^{-1}(K\tilde{\alpha}^t + \delta_{t+1} y_{t+1}\gamma^*_{t+1} J).$$

References

1. Kivinen J, Smola AJ, Williamson RC. Online learning with kernels. IEEE Trans Sig Process. 2004;52(8):2165–76.
2. Li GQ, Wen CY, Li ZG, Zhang A, Yang F, Mao K. Model-based online learning with kernels. IEEE Trans Neural Netw Learn Syst. 2013;24(3):356–69.
3. Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. J Mach Learn Res. 2011;12:2121–59.
4. Huang KZ, Yang HQ, Lyu MR. Machine learning: modeling data locally and globally. Springer Science & Business Media. 2008.
5. Orabona F, Keshet J, Caputo B. Bounded kernel-based online learning. J Mach Learn Res. 2009;10:2643–66.
6. Ertekin S, Bottou L, Giles CL. Nonconvex online support vector machines. IEEE Trans Pattern Anal Mach Intell. 2011;33(2):368–81.
7. Hoi SC, Wang JL, Zhao PL. Libol: A library for online learning algorithms. J Mach Learn Res. 2014;15(1):495–9.
8. Ding S, Zhang J, Jia H, Qian J. An adaptive density data stream clustering algorithm. Cogn Comput. 2016;8(1):30–8.
9. Gepperth A, Karaoguz C. A bio-inspired incremental learning architecture for applied perceptual problems. Cogn Comput. 2016;8(5):924–34.
10. Zhao J, Du C, Sun H, Liu X, Sun J. Biologically motivated model for outdoor scene classification. Cogn Comput. 2015;7(1):20–33.
11. Wang D, Qiao H, Zhang B, Wang M. Online support vector machine based on convex Hull vertices selection. IEEE Trans Neural Netw Learn Syst. 2013;24(4):593–609.
12. Ding SG, Nie XL, Qiao H, Zhang B. Online classification for SAR target recognition based on SVM and approximate convex hull vertices selection. In: 11th World Congress on intelligent control and automation (WCICA); 2014. p. 1473–1478.
13. Wu PC, Hoi SC, Zhao PL, Xia H, Liu ZY, Miao CY. Online multi-modal distance metric learning with application to image retrieval. IEEE Trans Knowl Data Eng. 2016;28(2):454–67.
14. Scardapane S, Uncini A. Semi-supervised echo state networks for audio classification. Cogn Comput. 2016;1–11.
15. Zhang YM, Huang KZ, Geng GG, Liu CL. A fast and robust graph-based transductive learning method. IEEE Trans Neural Netw Learn Syst. 2015;26(9):1979–91.
16. Zhu XJ, Rogers T, Qian RC, Kalish C. Humans perform semi-supervised classification too. In: Proceedings of the national conference on artificial intelligence. vol. 22. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999; 2007. p. 864.
17. Yang HQ, Huang KZ, King I, Lyu MR. Maximum margin semi-supervised learning with irrelevant data. Neural Netw. 2015;70:90–102.
18. Gibson BR, Rogers TT, Zhu XJ. Human semi-supervised learning. Topics Cogn Sci. 2013;5(1):132–72.
19. Babenko B, Yang MH, Belongie S. Visual tracking with online multiple instance learning. In: IEEE Conference on computer vision and pattern recognition; 2009. p. 983–990.
20. Grabner H, Leistner C, Bischof H. Semi-supervised on-line boosting for robust tracking. In: Computer Vision–European conference on computer vision. Springer; 2008. p. 234–247.
21. Dyer KB, Capo R, Polikar R. Compose: a semisupervised learning framework for initially labeled nonstationary streaming data. IEEE Trans Neural Netw Learn Syst. 2014;25(1):12–26.
22. Kveton B, Philipose M, Valko M, Huang L. Online semi-supervised perception: Real-time learning without explicit feedback. In: IEEE Computer society conference on computer vision and pattern recognition workshops (CVPRW); 2010. p. 15–21.
23. Farajtabar M, Shaban A, Rabiee HR, Rohban MH. Manifold coarse graining for online semi-supervised learning. In: Machine Learning and Knowledge Discovery in Databases. Springer; 2011. p. 391–406.

24. Goldberg AB, Li M, Zhu XJ. Online manifold regularization: A new learning setting and empirical study. Springer. 2008;393–407.
25. Goldberg AB, Zhu XJ, Furger A, Xu JM. OASIS: Online active semi-supervised learning. In: Proceedings of the Twenty-Fifth AAAI conference on artificial intelligence; 2011.
26. Sun BL, Li GH, Jia L, Zhang H. Online manifold regularization by dual ascending procedure. Math Probl Eng. 2013;2013.
27. Sun BL, Li GH, Jia L, Huang KH. Online coregularization for multiview semisupervised learning. Sci World J. 2013;2013.
28. Ding SG, Xi XY, Liu ZY, Qiao H, Zhang B. A novel manifold regularized online semi-supervised learning algorithm. In: International conference on neural information processing. Springer; 2016. p. 597–605.
29. Slater M. Lagrange multipliers revisited. Springer. 2014.
30. Belkin M, Niyogi P, Sindhwani V. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res. 2006;7:2399–434.
31. Schölkopf B, Herbrich R, Smola AJ. A generalized representer theorem. In: Computational learning theory. Springer; 2001. p. 416–426.
32. Melacci S, Belkin M. Laplacian support vector machines trained in the primal. J Mach Learn Res. 2011;12:1149–84.
33. Dekel O, Shalev-Shwartz S, Singer Y. The forgetron: A kernel-based perceptron on a budget. SIAM J Comput. 2008;37(5):1342–72.
34. Griva I, Nash SG, Sofer A. Linear and nonlinear optimization. 2009.
35. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324.
36. Heisele B, Poggio T, Pontil M. Face detection in still gray images. AI Memo 1697 Massachusetts Institute of Technology. 2000.
37. Liu J, Luo J, Shah M. Recognizing realistic actions from videos "in the wild". In: IEEE Conference on computer vision and pattern recognition, 2009. CVPR 2009. IEEE; 2009. p. 1996–2003.
38. Wang H, Kläser A, Schmid C, Liu CL. Action recognition by dense trajectories. In: 2011 IEEE Conference on computer vision and pattern recognition (CVPR). IEEE; 2011. p. 3169–3176.