LEARNING MULTI-GRAPH REGULARIZATION FOR SVM CLASSIFICATION

Vasileios Mygdalis, Anastasios Tefas and Ioannis Pitas

Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, 54124, Greece
Email: {tefas,pitas}@aiia.csd.auth.gr

ABSTRACT

A classification method that emphasizes on learning the hyperplane that separates the training data with the maximum margin in a regularized space, is presented. In the proposed method, this regularized space is derived by exploiting multiple graph structures, in the SVM optimization process. Each of the employed graph structure carries some information concerning a geometric or semantic property about the training data, e.g., local neighborhood area and global geometric data relationships. The proposed method introduces information from each graph type to the standard SVM objective, as a projection of the SVM hyperplane to such a direction, where a specific property of the training data is highlighted. We show that each data property can be encoded in a regularized kernel matrix. Finally, response in the optimal classification space can be obtained by exploiting a weighted combination of multiple regularized kernel matrices. Experimental results in face recognition and object classification denote the effectiveness of the proposed method.

Index Terms— Regularized Support Vector Machines, face recognition, object recognition

1. INTRODUCTION

Face recognition and object recognition are classic computer vision/pattern recognition classification problems that have found numerous applications in industrial and research fields, involving more and more sectors such as biometrics, virtual reality, and more lately applied robotics e.g., unmanned aerial vehicles. For the past two decades, the SVM classifier has been one of the most important baseline methods for tackling these tasks. The effectiveness of any classification method including the SVM classifier, depends on three things: a) the employed feature data representation, b) the suitability of the adopted kernel function to the employed features and finally c) the classifier discriminating ability.

Regarding features, recent work exploiting deep neural network architectures have revolutionized the efficiency of the obtained data representation in several problems including face [1, 2] and object recognition [3]. The resulting representation obtained by a forward-pass of the training data to specific layers of a pre-trained network can be used as feature vectors for detecting and recognizing objects depicted in images and videos. Moreover, work on obtaining improved kernel functions using Deep or Multiple Kernel Learning [4, 5, 6] have been performed with promising results. However, while feature vectors and kernel functions are rapidly improving the classification potential of the standard SVM classifier, we argue that its discriminating ability can be improved even further, by exploiting additional criteria in its optimization process.

Our argument is supported by relevant research on SVM-based methods, where it has been consistently shown that the SVM classification performance can be enhanced, while keeping the same initial feature data representation. This can be achieved by incrorporating additional optimization criteria to the SVM optimization process, e.g., discriminant learning [7, 8, 9], manifold learning [10], or as shown more recently, any geometric or semantic criteria that can modeled using generic graph structures [11, 12, 13, 14]. The employed graph structures that could be expressing intrinsic (within-class), or between-class data relationships, have the effect of promoting solutions in directions where the expressed property is emphasized (e.g., low-variance directions). Motivated by the success of graph-based SVM methods, we study the impact of introducing more than one or two properties to the SVM optimization problem.

In this paper, a classification method based on SVM, that learns the hyperplane that separates the training data with the maximum margin in a regularized space, where various data properties are emphasized, is proposed. This regularized space is derived by exploiting multiple graph structures in the SVM optimization process. Each of the employed graph carries some information concerning a geometric or semantic property about the training data, e.g., global variance, local neighborhood area. It is shown that each data property can be encoded in a regularized kernel matrix. The optimal classification space can be obtained by exploiting a weighted combination of multiple regularized kernel matrices. Therefore, the proposed optimization problem can be modeled as a standard Multiple Kernel Learning problem.

This work has received funding from the European Union's European Union Seventh Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE). This publication reflects only the authors' views. The European Commission is not responsible for any use that may be made of the information it contains.

2. MODELING DATA RELATIONSHIPS IN GRAPH STRUCTURES

Assume a dataset consisting of $d-$dimensional vectors $\boldsymbol{x}_i \in \mathbb{R}^D, i = 1, \ldots, N$ and $\boldsymbol{X} \in \mathbb{R}^{D \times N}$ is the corresponding data-matrix, that is employed to train a classifier. Pairwise properties within the training data can be expressed with set of graph structures $\mathcal{G}^m = \{\boldsymbol{X}, \boldsymbol{W}^m\}, m = 1, \ldots, M$, where the training data $\boldsymbol{X}$ form the graph vertices and the $m-$th graph expresses the $m-$th property about the training data. The matrix $\boldsymbol{W}^m$ is the corresponding graph weight matrix, containing the weights of the connections between the training data for the $m-$th property. Such pairwise properties may include e.g., local geometric data information expressed by kNN graphs, or global geometric data information expressed by fully connected graphs.

For example, the neighborhood $\mathcal{N}_i$ of each vertex could be containing the k most similar vectors to $\boldsymbol{x}_i$, and the weights between the vertices $\boldsymbol{W}_{ij}^l$ have been initiated with the following heat kernel function:

$$W_{ij}^l = \begin{cases} exp\left(-\gamma ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2\right), & \text{if } \boldsymbol{x}_j \in \mathcal{N}_i \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where γ is a free parameter that scales the Euclidean distances between the graph vertices $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. The, the local geometry of the training data is expressed with the matrix $\boldsymbol{S}_l$:

$$\boldsymbol{S}_l = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 W_{ij}^l = \boldsymbol{X} \boldsymbol{L}_l \boldsymbol{X}^T, \quad (2)$$

where $\boldsymbol{L}_l \in \mathbb{R}^{N \times N}$ is the corresponding graph Laplacian matrix, defined by $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D} \in \mathbb{R}^{N \times N}$ is the weighted graph Degree matrix, i.e., a diagonal matrix having elements $D_{ii} = \sum_{j \neq i} W_{ij}, i = 1, \ldots, N$.

The global geometry of the training data can be modeled by extending the above defined graph along all training data, i.e., by employing a fully connected graph ($k = N$). Alternatively, from a disciminant analysis perspective [15], a definition of global geometric data relationships is the following. Items belonging to the same class (e.g., class $c, c = 1, \ldots, C$) are connected in a fully connected graph $\mathcal{G}^w$ with equal weights, as follows:

$$W_{ij}^w = 1/N_c, \text{ if } y_i = y_j \quad (3)$$

where N_c is the number of items belonging to the $c-$th class. In fact, the corresponding matrix $\boldsymbol{S}_w$ that expresses global geometric data relationships as in equation (2), is the within-class scatter matrix:

$$\boldsymbol{S}_w = \sum_{c=1}^{C} \sum_{i=1}^{N_c} (\boldsymbol{x}_i^c - \bar{\boldsymbol{x}^c})(\boldsymbol{x}_i^c - \bar{\boldsymbol{x}}^c)^T =$$

$$= \boldsymbol{X} \left(\boldsymbol{I} - \sum_{c=1}^{C} \frac{1}{N_c} \boldsymbol{e}_c \boldsymbol{e}_c^T \right) \boldsymbol{X}^T = \boldsymbol{X} \boldsymbol{L}_w \boldsymbol{X}^T, \quad (4)$$

where $\boldsymbol{L}_w$ is the corresponding graph Laplacian matrix, c is an index denoting if $\boldsymbol{x}_i$ belongs to $c-$th class, i.e., $y_i = c$, and $\boldsymbol{e}_c$ is a vector of ones corresponding to the positions where $y_i = c$, or zeros, otherwise.

3. LEARNING MULTIGRAPH REGULARIZATION FOR SVM

The proposed method aims at generating the hyperplane $\boldsymbol{w}$ that separates the training data with the maximum margin, in a space where data relationships expressed with multiple graphs have been emphasized. The space where each property is emphasized can be implicitly obtained, by combining SVM hyperplanes $\boldsymbol{w}_m$ that have been regularized with the corresponding matrix $\boldsymbol{S}_m, m = 1, \ldots, M$, encoding information from the m-th graph. Moreover, since we would like to exploit multiple data properties, we also demand to learn their combination in an optimal manner. To this end, we introduce a vector $\boldsymbol{\mu} \in \mathbb{R}^M$, such that $\sum_{m=1}^{M} \mu_m = 1$, that controls the contribution of each data property to the final regularization effect. Therefore, the we propose to optimize for each hyperplane $\boldsymbol{w}_m$ and the parameters μ_m, at the same time, as follows:

$$\min_{\{\boldsymbol{w}\}, \boldsymbol{\xi}, b, \boldsymbol{\mu}} \quad \frac{1}{2} \sum_{m=1}^{M} \frac{1}{\mu_m} \left(\|\boldsymbol{w}_m\|^2 + \boldsymbol{w}_m^T \boldsymbol{S}_m \boldsymbol{w}_m \right) + c \sum_{i=1}^{N} \xi_i + b, \quad (5)$$

$$\text{s.t.} \quad y_i \left(\boldsymbol{w}_m^T \boldsymbol{x}_i + b \right) \leq 1 - \xi_i, i = 1, \ldots, N,$$

$$\xi_i \geq 0, \sum_{m=1}^{M} \mu_m = 1,$$

where ξ_i are the slack variables, b is the bias term and $c > 0$ is the SVM hyperparameter that needs to be tuned in order to provide the best compromise between training error and generalization performance (where a value $c = 0$ corresponds to hard margin SVM).

The above defined optimization problem can be solved by obtaining the equivalent dual problem, expressed with the Lagrangian function L and the Lagrange multipliers α_i corresponding to its constraints. By setting the partial derivatives of L with respect to $\boldsymbol{w}_m, \boldsymbol{\xi}, b$ equal to zero, and then by replacing back in L, we obtain an optimization problem of the following form:

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j q_{ij}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq c \quad (6)$$

where q_{ij} are the elements of a matrix $\boldsymbol{Q} \in \mathbb{R}^{N \times N}$, that contains the data similarity between the training data in the regularized space, as follows:

$$\boldsymbol{Q} = \boldsymbol{x}_i^T \left(\sum_{m=1}^{M} \mu_m \left(\boldsymbol{I} + \boldsymbol{S}_m \right)^{-1} \right) \boldsymbol{x}_j \quad (7)$$

or equivalently:

$$\boldsymbol{Q} = \left(\sum_{m=1}^{M} \mu_m \boldsymbol{x}_i^T \left(\boldsymbol{I} + \boldsymbol{S}_m \right)^{-1} \boldsymbol{x}_j \right) = \sum_{m=1}^{M} \mu_m \tilde{\kappa}_m(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{8}$$

where $\tilde{\boldsymbol{K}}_m$ is a similarity matrix that contains data similarity in the space where only the $m-$th data property have expressed. As can be observed, the proposed optimization problem can thereby be expressed as a Multiple Kernel Learning SVM problem:

$$\max_{\boldsymbol{\alpha}} \ \min_{\boldsymbol{\mu}} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \sum_{m=1}^{M} \mu_m \tilde{\kappa}_m(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{9}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq c \ \text{ and } \ \sum_{m=1}^{M} \mu_m = 1.$$

Solutions in the kernel space can be obtained by replacing the standard data representations $\boldsymbol{x}_i$ with the outputs of a mapping function $\phi(\boldsymbol{x}_i) \mapsto \mathcal{F}$. Although the data representations in $\mathcal{F}$ are of arbitrary dimensionality, only similarity in that space is required to be calculated explicitly for SVM classification purposes. By working in spaces of arbitrary dimensionality, the matrices $\boldsymbol{S}_m$ become of arbitrary dimensionality as well. Thankfully, the inversion $\left(\boldsymbol{I} + \boldsymbol{S}_m \right)^{-1}$ can still be obtained, by exploiting the Sherman-Morrison-Woodbury identity:

$$\left(\boldsymbol{I} + \boldsymbol{\Phi} \boldsymbol{L}_m \boldsymbol{\Phi}^T \right)^{-1} = \boldsymbol{I} - \boldsymbol{\Phi} \left(\boldsymbol{L}_m^{-1} + \boldsymbol{\Phi}^T \boldsymbol{\Phi} \right)^{-1} \boldsymbol{\Phi}^T, \tag{10}$$

where $\boldsymbol{\Phi} = [\phi(\boldsymbol{x}_1), \ldots, \phi(\boldsymbol{x}_N)]^T$ is a matrix that contains the data representations in the feature space. Similarity in the regularized space, that is required to be expressed with the matrices $\tilde{\boldsymbol{K}}_m$, is calculated explicitly as follows:

$$\tilde{\boldsymbol{K}} = \left[\boldsymbol{I} - \left(\boldsymbol{L}_m + \boldsymbol{K}^{-1} \right)^{-1} \boldsymbol{L}_m \right] \boldsymbol{K}, \tag{11}$$

where $\boldsymbol{K} = \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ is the standard kernel matrix obtained by e.g., applying the RBF kernel function.

In order to solve the optimization problem defined in (9), any Multiple Kernel Learning SVM method can be employed [5, 16]. We have employed [16] in all our experiments to this end, since it outperforms other widely adopted MKL methods e.g., [17, 18] in related classification problems, by providing an efficient compromise between sparse solutions and fast convergence. That is, the min-max optimization problem is broken into two quadratic programming optimization problems solved sequentially, one for the standard SVM, and a separate soft-margin optimization one for determining the parameters μ_m. For more information, the reader is referred to [16]. Finally, in order to classify a test sample $\boldsymbol{x}$, we employ the following decision function:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} y_i \alpha_i \sum_{m=1}^{M} \mu_m \tilde{\kappa}_m(\boldsymbol{x}_i, \boldsymbol{x}) + b. \tag{12}$$

The proposed method is able to include several graphs for regularization purposes, in the form of multiple single-graph regularized kernels. Thereby, the optimization problem is formulated as a MKL-SVM optimization problem. The advantage of this approach, is that it does not require fine tuning of the additional parameters that are introduced to control the amount of regularization. The effect of the each graph hyperparameters is implicitly determined only by optimally calculating the kernel contribution parameters μ_m, inside a separate optimization problem.

Additionally, the proposed formulation may serve as the general SVM formulation case, since related methods may be represented as special cases of the proposed method, by replacing the base kernel matrix inputs, with different matrices. For example, by replacing the derived kernel matrix $\boldsymbol{Q}$ with the standard SVM kernel matrix $\boldsymbol{K}$ and $\mu = 1$, the proposed method degenerates to standard SVM. By using a set of standard SVM kernel matrices derived by employing several mapping functions, or similar mapping functions with different parameters, the proposed method represents the generalized Multiple Kernel Learning [5] formulation. Finally, by introducing only a single graph in the SVM optimization process, the proposed method degenerates to the Graph-Embedded SVM [11] formulation.

4. EXPERIMENTS

In order to evaluate the performance of the proposed method, we have conducted experiments in publicly available datasets for face recognition and object classification. In our experiments in face recognition, we have employed the PubFig+LFW [19], AR [20], Yale [21] and ORL [22] datasets. In our object classification experiments, we have employed the CIFAR-100 [23] and Caltech101 [24] datasets. The employed datasets were carefully selected to demonstrate the effectiveness of the proposed method using different feature vector settings, i.e., pre-extracted feature vectors (PubFig+LFW), features having minimal preprocessing i.e., pixel luminosities (AR, Yale and ORL), pre-extracted deep features from [3] (CIFAR-100) and pre-computed kernel matrices [25] (Caltech101). Since all employed datasets are well balanced in terms of instances per class, for both training and testing purposes, the Classification Rate (CR) was employed as performance metric.

In order to construct our base kernels in all our experiments except Caltech101 (where we already had the precomputed kernels), we have employed the RBF$-\chi^2$ kernel matrix:

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|_2^2 \right), \tag{13}$$

where the optimal γ for every competing method was determined from a set of predefined values using cross validation. The regularized kernel matrices $\tilde{\boldsymbol{K}}$ were obtained using $\boldsymbol{S}_l$ and $\boldsymbol{S}_w$, as described in Section 2, for different values of γ.

For comparison reasons, along with the proposed method, we have also trained the standard Multiple Kernel Learning SVM (MKL-SVM) [16], using standard RBF kernels. Moreover, we also report the performance of the standard Graph-Embedded SVM (GE-SVM) [11], a standard SVM variant that employs a single regularized kernel matrix $\tilde{\boldsymbol{K}}$, which was in essence the best performing single base kernel of our proposed method. The same SVM solver was employed for all methods [26], and the parameter settings were also set to be equal for all methods, where applicable. Our experimental platform was a PC with 32GB of RAM on a i7 processor, using a Matlab implementation.

Experimental results are drawn in Tables 1 and 2 for face recognition and object classification, respectively. As can be seen, the proposed method outperformed the competing methods in every case, in terms of classification accuracy. More specifically, by observing the performance of all competing methods in PubFig+LFW dataset, whose feature vectors include information from hand-crafted descriptors, employing Multiple Kernel matrices seem to have been beneficial to classification performance. In such case, the extracted features lie in multiple distributions, and cannot be adequately modeled by a single distribution (i.e., the GE-SVM case). The performance of MKL-SVM denoted that exploiting multiple distributions for modeling data similarity was beneficial to performance in this case. The proposed method outperformed the competition, by exploiting the additional global and local geometric particularities of each class, modeled by the added graph structures. This information acted as an advanced regularizer to the solution, offering more accurate feature representation, in comparison with the competition.

In our experiments in ORL, YALE and AR, where we have employed simple pixel luminosities as feature vectors, we observed that employing multiple similar RBF kernel matrices, seem to have not influenced positively the classification performance, maybe related to overfitting issues. This effect is supported by the performance of GE-SVM, which outperformed MKL-SVM, by having a single graph regularizing the obtained classification space. However, the proposed method was able to alleviate the negative overfitting effects, by optimally determining the most efficient regularized kernel combination.

Finally, in our experiments in object recognition, the proposed method outpefromed the competition, again. Here we also note that the reported Caltech101 baseline MKL-SVM performance in [25] for the exact same kernels was 71.1%. Our employed baseline MKL-SVM method [16] outperformed this performance, obtaining a classification rate of 72.30%. Our proposed method outperformed MKL-SVM by 1.5%. This demonstrates the effectiveness of the proposed method, for the cases where deep features or pre-computed kernel matrices have been employed.

Table 1: Classification rates (CR) in Face Recognition

Algorithm/Dataset	PubFig+LFW	ORL	AR	Yale
GE-SVM	34.35	98.75	99.19	97.94
MKL-SVM	84.17	98.75	90.57	96.08
PROPOSED	**88.77**	**99.25**	**99.42**	**98.06**

Table 2: Classification rates (CR) in object recognition

Algorithm/Dataset	CIFAR-100	Caltech101
GE-SVM	72.30	66.56
MKL-SVM	75.40	72.42
PROPOSED	**79.80**	**73.39**

5. CONCLUSION

We described a novel method for introducing multiple pairwise data relationships represented in graph structures, to the SVM optimization process. The proposed optimization problem can be solved by exploiting generic Multiple-kernel learning SVM implementations. We have obtained increased classification performance consistently against related methods, in face recognition and object recognition classification problems. The reason for adopting increased performance is the exploitation of optimal kernel matrix regularization, specified for the SVM classification problem. Since the proposed method provided enhanced classification performance using different descriptor settings, we expect that it will perform well in other classification problems, as well.

6. REFERENCES

[1] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[2] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[3] Shuo Yang, Ping Luo, Chen Change Loy, Kenneth W Shum, Xiaoou Tang, et al., "Deep representation learning with target coding.," *In: AAAI Conference on Artificial Intelligence*, pp. 3848–3854, 2015.

[4] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P. Xing, "Deep kernel learning," in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, Arthur Gretton and Christian C. Robert, Eds., Cadiz, Spain, 09–11 May 2016, vol. 51 of *Proceedings of Machine Learning Research*, pp. 370–378, PMLR.

[5] Mehmet Gönen and Ethem Alpaydın, "Multiple kernel learning algorithms," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2211–2268, 2011.

[6] Serhat S Bucak, Rong Jin, and Anil K Jain, "Multiple kernel learning for visual object recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1354–1369, 2014.

[7] Dacheng Tao, Xuelong Li, Xindong Wu, and Stephen J Maybank, "Geometric mean for subspace selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 260–274, 2009.

[8] Stefanos Zafeiriou, Anastasios Tefas, and Ioannis Pitas, "Minimum class variance support vector machines," *IEEE Transactions on Image Processing*, vol. 16, no. 10, pp. 2551–2564, 2007.

[9] Naimul Mefraz Khan, Riadh Ksantini, Imran Shafiq Ahmad, and Ling Guan, "Covariance-guided one-class support vector machine," *Pattern Recognition*, vol. 47, no. 6, pp. 2165–2177, 2014.

[10] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of machine learning research*, vol. 7, no. Nov, pp. 2399–2434, 2006.

[11] Georgios Arvanitidis and Anastasios Tefas, "Exploiting graph embedding in support vector machines," *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2012.

[12] Vasileios Mygdalis, Alexandros Iosifidis, Anastasios Tefas, and Ioannis Pitas, "Graph embedded one-class classifiers for media data classification," *Pattern Recognition*, vol. 60, pp. 585 – 595, 2016.

[13] Alexandros Iosifidis and Moncef Gabbouj, "Multi-class support vector machine classifiers using intrinsic and penalty graphs," *Pattern Recognition*, vol. 55, pp. 231–246, 2016.

[14] Vasileios Mygdalis, Alexandros Iosifidis, Anastasios Tefas, and Ioannis Pitas, "Semi-supervised subclass support vector data description for image and video classification," *Neurocomputing*, vol. 278, pp. 51 – 61, 2018.

[15] Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-Jiang Zhang, Qiang Yang, and Stephen Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[16] Xinxing Xu, Ivor W Tsang, and Dong Xu, "Soft margin multiple kernel learning," *IEEE transactions on neural networks and learning systems*, vol. 24, no. 5, pp. 749–761, 2013.

[17] Marius Kloft, Ulf Brefeld, Pavel Laskov, Klaus-Robert Müller, Alexander Zien, and Sören Sonnenburg, "Efficient and accurate lp-norm multiple kernel learning," *Advances in neural information processing systems*, pp. 997–1005, 2009.

[18] Manik Varma and Bodla Rakesh Babu, "More generality in efficient multiple kernel learning," *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 1065–1072, 2009.

[19] Brian Becker and Enrique Ortiz, "Evaluating open-universe face identification on the web," *In: Computer Vision and Pattern Recognition (CVPR)*, pp. 904–911, 2013.

[20] Aleix M Martinez, "The ar face database," *In: CVC Technical Report*, vol. 24, 1998.

[21] Athinodoros S Georghiades, Peter N Belhumeur, and David J Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[22] F.S. Samaria and A.C. Harter, "Parameterisation of a stochastic model for human face identification," *In: IEEE Workshop on Applications of Computer Vision (1994)*, 1994.

[23] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," 2009.

[24] Li Fei-Fei, Rob Fergus, and Pietro Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer vision and Image understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[25] Andrea Vedaldi, Varun Gulshan, Manik Varma, and Andrew Zisserman, "Multiple kernels for object detection," *In: International Conference on Computer Vision*, pp. 606–613, 2009.

[26] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.