WEIGHTED SVM WITH CLASSIFICATION UNCERTAINTY FOR SMALL TRAINING SAMPLES

Min-Kook Choi *Hyun-Gyu Lee* *Sang-Chul Lee*

Department of Computer Engineering, Inha University, Korea
{*mkchoi*, *hg_lee*, *sclee*}@inha.edu

ABSTRACT

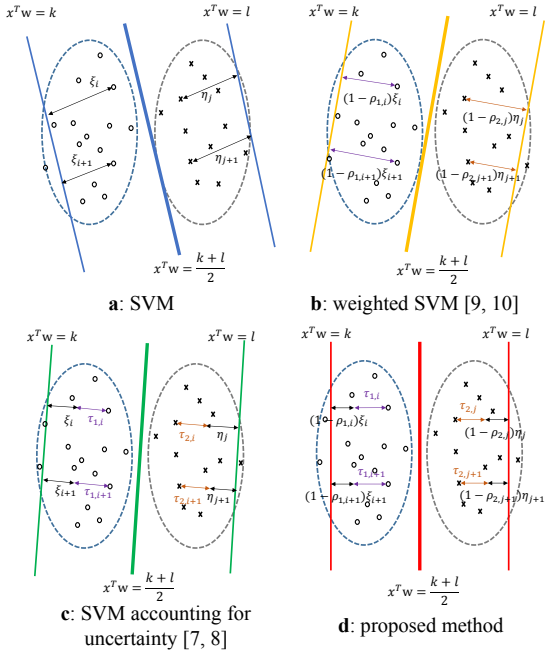
Existing maximum-margin support vector machines (SVMs) generate a hyperplane which produces the clearest separation between positive and negative feature vectors. These SVMs are effective when datasets are large. However, when few training samples are available, the hyperplane is easily influenced by outliers that are geometrically located in the opposite class. We propose a modified SVM which weights feature vectors to reflect the local density of support vectors and quantifies classification uncertainty in terms of the local classification capability of each training sample. We derive a primal formulation of an SVM that incorporates those modifications, and implement an RC-margin SVM of the simplest form. We evaluated our model on the recognition of handwritten numerals, and obtained higher recognition rate than a standard maximum-margin SVM, a weighted SVM, or an SVM which accounts for classification uncertainty.

Index Terms— Weighted SVM, Classification uncertainty, Reduced convex hulls

1. INTRODUCTION

The problem of discriminating between many classes when availability of training data is limited is a major challenge in machine vision. A maximum-margin classification criterion [1, 2] has been widely used to address this problem using a support vector machine (SVM). This approach has had remarkable success on many problems, but it has serious limitations such as object or action recognition, especially when training samples are small and have many outliers. Some research have tried to conduct a geometrically more meaningful separating criterion [3, 4, 5, 6, 7, 8]; others have introduced different penalties for misclassification of training samples [5, 9, 10, 11, 12]; and local approximation techniques [13, 14, 15] and semi-supervised learning [16, 17].

In this paper, we propose an integrated classification model which combines the maximum-margin criterion with an alternative penalty based on reduced convex hulls which has more meaningful geometric interpretation. The optimization penalty is computed by combining feature vector weights and classification uncertainties, and is obtained locally for each training feature vector. The combined penalty for each feature vector is obtained by considering: 1) the local density of the nearest feature vectors, and 2) the uncertainty of classification, based on the classification of local training samples. Figure 1 illustrates how the primal formulation of an SVM model, determined by its separating scheme and geometric interpretation of the misclassification penalty, affects



a: SVM **b**: weighted SVM [9, 10]

c: SVM accounting for uncertainty [7, 8] **d**: proposed method

Fig. 1. Examples of four different SVMs used to separate 2D points with a Gaussian distribution (dashed ellipse). $x^T w = k$ and $x^T w = l$ are two parallel bounding planes maximizing separation between point clouds. **a**. The max-margin SVM uses the slack variables ξ and η to apply misclassification penalties to the positive and negative classes. **b**. The weighted SVM imposes an additional weight $(1-\rho)$ on each slack variable. **c**. An SVM which accounts for uncertainty imposes an additional misclassification penalty τ on each training sample. **d**. Our method determines a misclassification penalty by combining the penalties of **b** and **c**.

on the separating hyperplane. Our formulation uses reduced convex hulls (RCHs) to construct a geometric interpretation of the different misclassification penalties (i.e. the classification weights applied to each training sample), and the local classification capability (i.e. the classification uncertainty within a group of training samples). The result is hyperplane optimization which is robust against noisy training data, and works well when the training dataset is small.

This work values two main contributions: 1) We use new geometric intuition to generate a separating hyperplane which is robust against noise and the effect of the limited number of training samples. This formulation is derived by applying the RC-margin SVM classifier to the modified SVM. The modified SVM reduces to the problem of solving the closest pair of points problem [3], and the optimized separating hyperplane is a perpendicular bisection or a line segment connecting the closest pair of points. This method can be extended to nonlinear case using a kernel mapping, and/or to multi-class classification problems by constructing pairwise classifiers. 2) The modified SVM performs better than the previous SVMs in multi-class classification for machine vision application with sparse training samples. We assessed the effectiveness of our model in the problem of handwritten numeral recognition from the MNIST database [18]. Our model achieved approximately 8% higher recognition rate than a conventional SVM in digit recognition with limited training samples.

2. BACKGROUND

Primal form of the RC-margin SVM. We represent n items of training data in the non-separable form: positive class: $A_{p\times n_1} = [x_1, x_2, ..., x_{n_1}]$ and negative class: $B_{p\times n_2} = [x_1, x_2, ..., x_{n_2}]$, and so $n_1 + n_2 = n$. Bennett and Bredensteiner [3] proposed different approaches in order to construct a separating hyperplane by finding the shortest path between RCHs. They used the primal optimization to find the shortest path between two RCHs with soft margins, which is written as follows:

$$\begin{aligned} \min_{w,\xi,\eta,k,l} \quad & \tfrac{1}{2}w^T w - k + l + C(\xi^T e + \eta^T e), \\ \text{s.t. } & A^T w - ke + \xi \geq 0, \quad \xi \geq 0, \\ & -B^T w + le + \eta \geq 0, \quad \eta \geq 0. \end{aligned} \tag{1}$$

where k and l are the offsets of the separating hyperplane $x^T w = (k + l)/2$, the slack variables $\xi_{1\times n_1}$ and $\eta_{1\times n_2}$ provide soft margins, e is a vector with one dimension for each element, and C is a weighting parameter for reducing the convex hull. The valid range of C is typically $\frac{1}{M} \leq C \leq 1$, where $M = min(n_1, n_2)$. A value of C larger than 1 provides no advantage in optimization; and if C is less than $\frac{1}{M}$, there is no feasible solution in the feature space. The RC-margin optimization [3] described by Equation (1) is a baseline optimization which provides the concepts needed to integrate geometric intuition into our SVM. We now describe how to derive the dual optimization of our SVM from the primal optimization of SVMs that uses geometric variants of the max-margin to generate different hyperplanes, including a weighted SVM [11] and an RC-margin SVM with classification uncertainty [5, 10].

Weighted RC-margin SVM. We use weights that supplement the misclassification penalty, derived from the geometry of each training sample. Since such a geometric penalty punishes outliers, this scheme is more effective when there is a limited amount of training data. We define a weight vector ρ_y, in which element $\rho_{(y,i)}$ weights the i^{th} training sample in class y. The primal optimization of this weighted RC-margin SVM can now be derived from Equation (1) as follows:

$$\begin{aligned} \min_{w,\xi,\eta,k,l} \quad & \tfrac{1}{2}w^T w - k + l + D(\xi^T(e - \rho_1) + \eta^T(e - \rho_2)), \\ \text{s.t. } & A^T w - ke + \xi \geq 0, \quad \xi \geq 0, \\ & -B^T w + le + \eta \geq 0, \quad \eta \geq 0. \end{aligned} \tag{2}$$

where $\rho_1 \in \mathbb{R}^{n_1\times 1}$ and $\rho_2 \in \mathbb{R}^{n_2\times 1}$ are weight vectors that satisfy the normalization conditions $\sum_{i=1}^{n_1} \rho_{1,i} = 1$ and $\sum_{i=1}^{n_2} \rho_{2,i} = 1$. The weighting parameter D reduces the hyper-volume enclosed by the convex hull. The valid range of D is $\frac{1}{M} \leq D \leq 1$.

RC-margin SVM with classification uncertainty. We define a classification uncertainty vector τ_y, where element $\tau_{(y,i)}$ is the uncertainty of the i^{th} training sample in the class y, which is measured by its distance from the opposite class, determined by a linear classifier. The primal optimization of the RC-margin SVM with classification uncertainty can be derived from Equation (1) as follows:

$$\begin{aligned} \min_{w,\xi,\eta,k,l} \quad & \tfrac{1}{2}w^T w - k + l + E(\xi^T e + \eta^T e), \\ \text{s.t. } & A^T w + \tau_1 - ke + \xi \geq 0, \quad \xi \geq 0, \\ & -B^T w + \tau_2 + le + \eta \geq 0, \quad \eta \geq 0. \end{aligned} \tag{3}$$

where τ_1 and τ_2 are classification uncertainty vectors of dimension $n_1 \times 1$ and $n_2 \times 1$, respectively. The weighting parameter E is reduced to the size of convex hull, and the valid range of E is $\frac{1}{M} \leq E \leq 1$.

3. WEIGHTED RC-MARGIN SVM WITH CLASSIFICATION UNCERTAINTY

We now proceed to formulate an RC-margin SVM where the geometric weights and the uncertainty of classification are combined. Using the Equations (2) and (3), two parallel bounding planes are defined in a primal form:

$$\begin{aligned} \min_{w,\xi,\eta,k,l} \quad & \tfrac{1}{2}w^T w - k + l + Q(\xi^T(e - \rho_1) + \eta^T(e - \rho_2)), \\ \text{s.t. } & A^T w + \tau_1 - ke + \xi \geq 0, \quad \xi \geq 0, \\ & -B^T w + \tau_2 + le + \eta \geq 0, \quad \eta \geq 0. \end{aligned} \tag{4}$$

where Q is a weighting parameter to reduce the size of the convex hull, and the valid range of Q is $\frac{1}{M} \leq Q \leq 1$. To solve this primal optimization, we impose non-negative Lagrangian multipliers $\mu_{n_1 \times 1}, \gamma_{n_1 \times 1}, \nu_{n_2 \times 1}$, and $\zeta_{n_2 \times 1}$, and then take partial derivatives of L with respect to the each primal variable. Thus we can formulate the optimization as follows:

$$\begin{aligned} &\min_{w,\xi,\eta,\mu,\gamma,\nu,\zeta,k,l} L(w,\xi,\eta,\mu,\gamma,\nu,\zeta,k,l) \\ &= \tfrac{1}{2} w^T w - k + l + Q(\xi^T(e-\rho_1) + \eta^T(e-\rho_2)) \\ &-\mu^T(A^T w + \tau_1 - ke + \xi) \\ &-\nu^T(-B^T w + \tau_2 - le + \eta) - \gamma^T \xi - \zeta^T \eta. \end{aligned} \tag{5}$$

$$\begin{aligned} \text{s.t. } \frac{\partial L}{\partial w} &= w - A^T\mu + B^T\nu = 0, \\ \partial L/\partial k &= -1 + \mu^T e = 0, \quad \mu \geq 0, \\ \partial L/\partial l &= 1 - \nu^T e = 0, \quad \nu \geq 0, \\ \partial L/\partial \xi &= Q(1-\rho_1) - \mu = \gamma \geq 0, \\ \partial L/\partial \eta &= Q(1-\rho_2) - \nu = \zeta \geq 0. \end{aligned} \tag{6}$$

where μ, γ, ν, and ζ are non-negative Lagrangian multipliers, which are subject to the constraints that their partial derivatives with respect to the primal variables are equal to zero. To simplify Equations (5) and (6), we eliminate $w, k, l, \xi, \eta, \mu, \nu, \gamma$, and ζ by substituting in $w = A^T\mu - B^T\nu$, $\gamma = Q\rho_1 - \mu$ and $\zeta = Q\rho_2 - \nu$. Then we obtain a dual form that expresses the shortest path between the reduced convex hulls:

$$\begin{aligned} &\max_{w,\xi,\eta,k,l} -\tfrac{1}{2}\|A^T\mu - B^T\nu\|^2 - (\tau_1^T\mu + \tau_2^T\nu), \\ &\text{s.t. } \mu^T e - 1 = 0 \quad 1 - \nu^T e = 0, \\ &0 \leq (1-\rho_{1,i})\mu_i \leq Q, \quad 0 \leq (1-\rho_{2,i})\nu_i \leq Q. \end{aligned} \tag{7}$$

where $A^T\mu$ and $B^T\nu$ represent feature vectors in the convex hulls of the positive and the negative classes, and μ and ν are combined coefficients which are applied to each feature vector inside the convex hulls. The only free parameter Q is an upper bound on the weighted combined coefficients $(1-\rho_{1,i})\mu_i$ and $(1-\rho_{2,i})\nu_i$. Q is equivalent to the optimization regularizer λ in the max-margin SVM.

Weights and classification uncertainties. The weight $\rho_{1,i}$ applied to the i^{th} feature vector in the positive class is set to the average of the normalized L_2 distances between the i^{th} feature vector and the h_w nearest feature vectors:

$$\rho_{1,i} = \frac{1}{h_w}\sum_{k=1}^{h_w} d(x_i, x_{\dot{k}}), \quad i \neq \dot{k}. \tag{8}$$

where d is the L_2 distance between the i^{th} feature vector and k^{th} nearest feature vectors from the i^{th} feature vector. Finding the nearest feature vectors can be expensive if h_w is large. Therefore we construct a non-linear embedding [19], which allows the h_w nearest feature vectors to be found quickly. The weights $\rho_{2,i}$ applied to the feature vectors in the negative class are determined in a similar way, and ρ_1 and ρ_2 are separately normalized to the range 0 to 1.

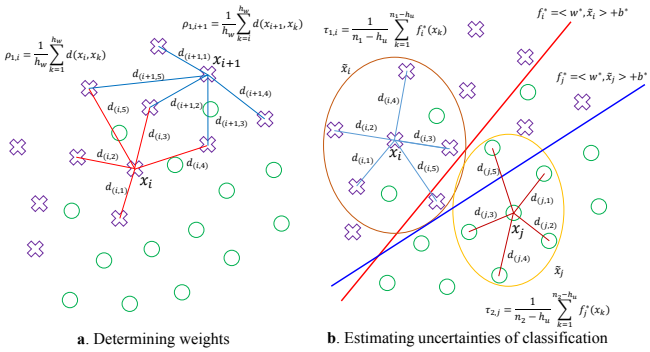


Fig. 2. Determining weights and classification uncertainties when h_w and h_u are set to 5. **a**. The L_2 distance of training samples is d, and $\rho_{1,i}$ and $\rho_{2,i}$ are the resulting weights. **b**. The local linear classifier f_i^* operates on the 5 nearest samples $\tilde{x}_i$ to measure the classification capability of a training sample x_i. The resulting estimates of uncertainty are $\tau_{1,i}$ and $\tau_{2,j}$. $d_{(i,\dot{k})}$ denotes the distance between x_i and the k^{th} closest feature vector to x_i.

The uncertainty of the i^{th} feature vector x_i, written as τ_i, is estimated by a similar method to that of [5] where the classification capability of a feature vector is measured by running a local linear SVM classifier $f_i^* =< w^*, \tilde{x} > +b^*$, where $\tilde{x}$ is a set of feature vectors consisting of the vector x_i and its h_u nearest neighbors. For the i^{th} feature vector, this classifier uses the h_u nearest feature vectors of the positive examples in the class for the training. The uncertainty of the i^{th} feature vector x_i, which is placed in the positive class, is estimated by the proportion of correct classifications of the local linear classifier:

$$\tau_{1,i} = \frac{1}{n_1 - h_u}\sum_{k=1}^{n_1 - h_u} f_i^*(x_k). \tag{9}$$

The uncertainty of the j^{th} feature vector x_j, which is placed in the negative class, is estimated in a similar way. The uncertainties τ_1 and τ_2 are normalized separately to the range 0 to 1, which are founded by non-linear embedding [19]. Figure 2 illustrates these methods of determining the weight and classification uncertainty for some feature vectors.

4. EXPERIMENTAL RESULTS

Parameter selection. We performed a series of experiments using [18] to investigate how well our SVM works with different parameters, and compared the results with the classic and weighted SVM techniques and with recent SVM that considers classification uncertainty [5, 11]. We implemented our model using the libSVM library [20]. Our model has a parameter Q which weights the reduction of convex hulls. The external parameter h_w determines how many nearest feature

vectors are used to obtain the local density ρ; and h_u determines the number of nearest feature vectors used to construct the local linear classifier that estimates the classification uncertainty vector τ.

The valid range of Q is $[\frac{1}{min(n_1,n_2)}, 1]$. Figure 3 **a** shows the accuracy of digit recognition with different numbers of training samples and different values of h_w and h_u, and Figure 3 **b** shows the results for different values of Q. As a result of these experiments, we set Q to 0.9, h_w to 9 and h_u to 15. Model parameters D and h_w for the weight SVM, and E and h_u for the SVM which accounts for classification uncertainty are tuned in the same way. We selected the final parameters D to 0.85, h_w to 7, E to 0.83, and h_u to 12.

Recognition rate. We used four types of SVM models with different numbers of training samples to recognize digits from the MNIST database which contains $70,000$ handwritten numerals from 0 to 9, where $60,000$ samples were used for training and $10,000$ samples for testing. Each numeral is stored as a single feature vector consisting of 784 normalized pixel values after L_2 normalization. Classification errors were evaluated using the test set after training the SVM classifier with 100 samples.

Figure 4 **a** shows classification accuracies with different numbers of training samples, and Figure 4 **b** shows the variations in accuracy for each digit $(0-9)$, for the weighted SVM, an SVM with classification uncertainty, and our technique, compared with the standard SVM, for 200 training samples. This result shows that our technique produces higher recognition rate than existing techniques when few samples are available.

Computational complexity. The computational complexity of our SVM is dominated by the time required to find the nearest neighbors to determine the weights and to construct local classifiers to measure uncertainty in multi-class classification. The non-linear embedding technique used to find the nearest neighbors takes $O(np)$ for all training sample [19], and so the total time required to search for nearest neighbors is $O(h_w h_u np)$. Since h_w and h_u are small constants less than one hundred, the time required for the nearest neighbor search is effectively to linear. To generate the local classifiers, we extend our model to a multi-class problem by using one-against-one method for m classes. All the possible $(m-1)/2$ pairwise binary classifiers are constructed in order to determine the final class, which is the one that gathers the most pairwise classifications. The learning time of the final multi-class classification is $O(snm)$, where $s = h_w h_u p$.

5. CONCLUSION

We proposed a new SVM model where weighted vectors and a measure of classification uncertainty are combined with an RC-margin classifier. We derived the primal formula for this

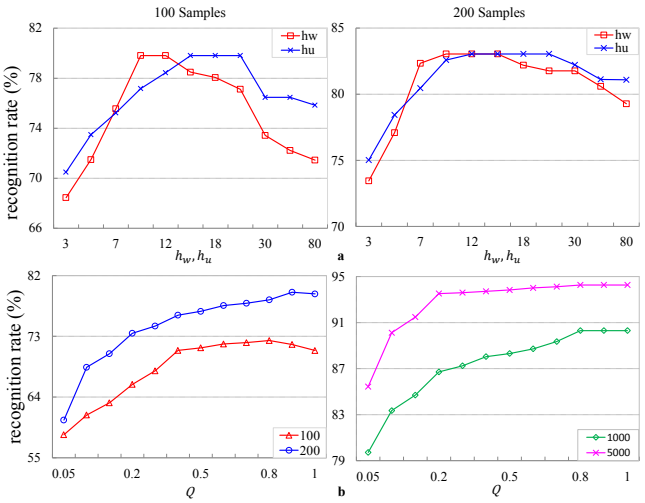


Fig. 3. Parameter tuning through the recognition of digits from the MNIST database. **a**. h_w and h_u are while Q is fixed at 0.9. **b**. Q is valid while $h_w = 9$ and $h_u = 15$.

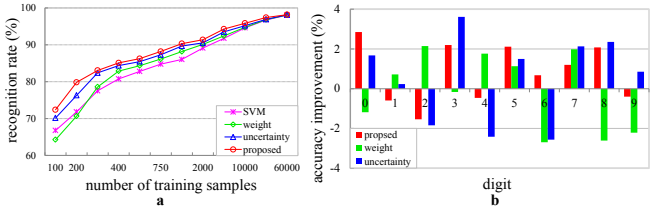


Fig. 4. Result from digit recognition: **a**. recognition rate for four different SVM models; **b**. comparison of the three modified SVM techniques with classic SVM, using 200 training samples.

SVM, and then solved a dual optimization problem. In our experiments, we demonstrated the effectiveness of this modified SVM in recognition of handwritten numerals from the MNIST database. Our SVM outperformed a standard SVM using the maximum-margin framework by 8%, when the number of training samples is relatively small. In future work we may extend our method to apply other applications, such as image classification and human action recognition. We also expect that a modified implementation based on linear approximations [21] would reduce training times.

6. ACKNOWLEDGEMENTS

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (No. 2012M3C4A7032781) and Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2015R1D1A1A01057968).

7. REFERENCES

[1] G. Bouchard and B. Triggs, "The tradeoff between generative and discriminative classifiers," in *Proc. of International Symposium on Computational Statistics*, 2004.

[2] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–197, 1995.

[3] K. P. Bennett and E. J. Bredensteiner, "Duality and geometry in svm classifiers," in *Proc. of International Conference on Machine Learning*, 2000.

[4] M. E. Mavroforakis and S. Theodoridis, "A geometric approach to support vector machine (svm) classification," *IEEE Transactions on Neural Networks*, vol. 17, no. 3, pp. 671–682, 2006.

[5] W. Zhang, S. X. Yu, and S. H. Teng, "Power svm: Generalization with exemplar classification uncertainty," in *Proc. of Computer Vision and Pattern Recognition*, 2012.

[6] X. Peng and Y. Wang, "Geometric algorithms to large margin classifier based on affine hulls," *IEEE Transactions on Neural Networks and Learning System*, vol. 23, no. 2, pp. 236–246, 2012.

[7] M. Carrasco, J. Lopez, and S. Maldonado, "A multiclass svm approach based on the l1-norm minimization of the distances between the reduced convex hulls," *Pattern Recognition*, vol. 48, pp. 1598–1607, 2015.

[8] M. Zeng, Y. Yang, J. Zheng, and J. Cheng, "Maximum margin classification based on flexible convex hulls," *Neurocomputing*, vol. 149, pp. 957–965, 2015.

[9] C. F. Lin and S. D. Wang, "Fuzzy support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 464–471, 2002.

[10] J. Bi and T. Zhang, "Support vector classification with input data uncertainty," in *Advanced in Neural Information Processing Systems*, 2004.

[11] Y. M. Huang and S. X. Du, "Weighted supoort vector machine for classification with uneven training class sizes," in *Proc. of International Conference on Machine Learning and Cybernetics*, 2005.

[12] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of exemplar-svms for object detection and beyond," in *Proc. International Conference on Computer Vision*, 2011.

[13] C. Domeniconi and D. Gunopulos, "Adaptive nearest neighbor classification using support vector machine," in *Advanced in Neural Information Processing Systems*, 2011.

[14] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "Svm-knn: Discriminative nearest neighbor classification for visual category recognition," in *Proc. of Computer Vision and Pattern Recognition*, 2006.

[15] S. Bengio, J. Weston, and D. Grangier, "Label embedding trees for large multi-class task," in *Advanced in Neural Information Processing Systems*, 2010.

[16] D. Zhou, O. Bousquet, J. Weston T. N. Lal, and B. Scholkopf, "Learning with local and global consistency," in *Advance in Neural Information Processing Systems*, 2004.

[17] R. Fergus, Y.Weiss, and A. Torralba, "Semi-supervised learning in gigantic image collections," in *Advanced in Neural Information Processing Systems*, 2009.

[18] Y. LeCun, L. Bottou, Y. Bengio, , and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. of the IEEE*, 1998.

[19] Y. Hwang, B. Han, and H. Ahn, "A fast nearest neighbor search algorithm by nonlinear embedding," in *Proc. of Computer Vision and Pattern Recognition*, 2012.

[20] C. C. Chang and C. J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 27, pp. 1–27, 2011.

[21] S. Maji, S. C. Berg, and J. Malik, "Classification using intersection kernel support vector machine is efficient," in *Proc. of Computer Vision and Pattern Recognition*, 2008.