21st International Conference on Pattern Recognition (ICPR 2012)
November 11-15, 2012. Tsukuba, Japan

KL Based Data Fusion for Target Tracking

Jing Peng
Montclair State University
Montclair, NJ 07003
jing.peng@montclair.edu

K. Palaniappan & Sema Candemir
University of Missouri
Columbia, MO 65211-2060
palaniappank@missouri.edu

Guna Seetharaman
AFRL/RIEA
Rome, NY 13441
guna@ieee.org

Abstract

Visual object tracking in video can be formulated as a time varying appearance-based binary classification problem. Tracking algorithms need to adapt to changes in both foreground object appearance as well as varying scene backgrounds. Fusing information from multimodal features (views or representations) typically enhances classification performance without increasing classifier complexity when image features are concatenated to form a high-dimensional vector. Combining these representative views to effectively exploit multimodal information for classification becomes a key issue. We show that the Kullback-Leibler (KL) divergence measure provides a framework that leads to family of techniques for fusing representations including Chernoff distance and variance ratio that is the same as linear discriminant analysis. We provide experimental results that corroborate well with our theoretical analysis.

1 Introduction

Classifiers employed in real world scenarios must deal with various adversities such as noise in sensors, intra-class variations, and restricted degrees of freedom. It is often helpful to develop classifiers that rely on data from various sources for classification. Such classifiers require an effective way of fusing the various sources of information. Resulting fused classifiers can offer a number of advantages, such as increased confidence in decision-making, resulting from fused complementary data, and robust performance against noise. Multi-view learning finds its applications in many domains such as defense, medicine, and sciences [11].

While algorithms that combine multi-view information are known to exponentially quicken object identification and classification, they lack the ability to seek out relevant information and update actionable conclusions to augment a decision process. In this paper, we propose a technique that computes optimal combination weight by minimizing an objective that measures the difference between an ideal (but unknown) probability and its estimate to address these problems. Here a representation (view) corresponds to a specific type of feature or attribute. For example, an image can be represented by (1) texture features, (2) edge features, and/or (3) shape features. Thus, each of these views provides a partial view about an object of interest, i.e., revealing a particular aspect of data. Our method provides a mechanism to exploit all of the data available, and as such, the method can be very useful for making inferences about potential objects of interest characterized with multiple views.

It turns out that the resulting weight is related to the degree of class separability that representation can provide. We show a number of practical ways to estimate class separability. We also provide experimental results that corroborate our theoretical analysis.

2 Related Work

In multi-view learning, a co-training procedure for classification problems was developed [3]. The idea is that better classifiers can be learned at the individual view level, rather than constructed directly on all the available views. The co-training procedure builds a separate classifier for each view and a final classifier can be built by combining the individual view classifiers in order to make prediction. The idea is that if the results of the individual classifiers arrive at the same conclusion for a given input, and the views are conditionally independent, then the classification is highly likely correct [1, 3].

Multi-view learning can also be addressed by defining a kernel for each view, and convexly combining the resulting kernels for classification [6, 18]. Related to the kernel idea, graph methods convexly combine graph Laplacians on different views for learning [14, 2].

In [15] stacked generalization was proposed. It is a

general technique for construction of multi-level learning systems. In the context of multi-view learning, it yields unbiased, full-size training sets for the trainable combiner. Stacked generalization is defined as any scheme that feeds data from one set of classifiers to another before making a final decision. In some cases stacked generalization is equivalent to cross-validation, in other cases it is equivalent to forming a linear combination of the classification results of the constituent classifiers. In [17], a local learning technique was proposed that combines multi-view information for better classification.

In many ways, multi-view learning and data fusion address the same set of problems. From the viewpoint of data fusion, comprehensive surveys of various classifier fusion studies and approaches can be found in [8, 10]. More recently, Lanckriet et al. [11] introduce a kernel-based data fusion (multi-view learning) approach to protein function prediction in yeast. The method combines multiple kernel representations in an optimal fashion by formulating the problem as a convex optimization problem that can be solved using semi-definite programming.

In [4], a technique is introduced that attempts to select the best likelihood maps that clearly separate object from background. In particular, linear combinations of (R, G, B) color space are mapped via likelihood ratio to generate candidate likelihood maps. A pooled variance ratio is used to rank each candidate in such a way that a candidate map is ranked high if it maximizes total target and background variance, while at the same time minimizing within class (predicted) variance.

In [16], a theoretical justification for the technique proposed in [4] is provided. It turns out that the combination coefficient for a likelihood map is inversely proportional to the degree of uncertainty associated with the map for predicting true target locations. There are two major weaknesses in this work. First, there is a gap between what the theory predicts and the variance ratio used in [4]. Second, it makes the assumption that estimates provided by likelihood maps are unbiased in justifying the variance ratio, which is highly unrealistic.

3 Problem Formulation

The goal of target tracking is to predict state variable x or the location of the target object from input image I, i.e. solving $P(x|I)$, the probability of x being the target location, given the input image I. We call $P(x|I)$ the desired probability. We have a set of M representations (features) that characterize the input image.

We can estimate the desired probability from each of these representations. Let us denote the estimate from the ith representation by $P(x|R_i)$. We call $P(x|R_i)$ the likelihood maps. Now our goal is to fuse these likelihood maps to best approximate $P(x|I)$. That is, we want to compute the optimal combination weight for the $P(x|R_i)$s so that the fused likelihood map is an accurate approximation to the desired probability $P(x|I)$.

We propose to explore the Kullback-Leibler (KL) divergence as our objective function [9]. The KL divergence measures the difference between two density distributions. The KL divergence is also known as relative entropy or information divergence. It can be precisely defined as the average of a log-likelihood ratio of two densities and it is the exponent in the theory of large deviations. We note that the KL divergence is zero when the two distributions are the same. It is greater than zero when they differ. Thus, it is appropriate for our purpose here.

4 Optimal Combination Weight

The objective is to predict variable x denoting the target object from input image I, i.e. solving $p(x|I)$. Given a set of representations R_i based on different feature extraction mechanisms, we apply Bayesian inference to obtain

$$p(x|I) = \int p(x|R)p(R|I)dL \approx \sum_i^M w_i p(x|R_i) \quad (1)$$

where M denotes the number of representations, and $w_i = p(R_i|I)$ Also, $\sum_i^M w_i = 1$. We note that here we have made the assumption that representations are independent given an input image.

The goal is to estimate the optimal weights w_i so that $\sum_i^M w_i p(x|R_i)$ will best approximate $p(x|I)$. Notice that there are two possible outcomes of x: ω_1, indicating that x is a target, and ω_2, indicating that x is non-target. We optimize the following KL divergence between $p(x|I)$ and $\sum_i^M w_i p(x|R_i)$

$$J(w) = \sum_{x \in \{\omega_1, \omega_2\}} p(x|I) \log(\frac{p(x|I)}{\sum_i^M w_i p(x|R_i)}) \quad (2)$$

subject to $w^t \mathbf{1} = 1$, where $w = (w_1, w_2, \cdots, w_M)^t$, and $\mathbf{1} = (1, 1, \cdots, 1)^t$.

If we write $p_{\omega_i} = p(x = \omega_i|I)$ and $P_{\omega_i} = (p(x = \omega_i|R_1), p(x = \omega_i|R_2), \cdots, p(x = \omega_i|R_M))^t$, where $i = 1, 2$. It follows that the unconstraint optimization problem becomes

$$J(w, \lambda) = \sum_i p_{\omega_i} \log(\frac{p_{\omega_i}}{w^t P_{\omega_i}}) + +\lambda(w^t \mathbf{1} - 1). \quad (3)$$

Differentiating both sides with respect to w and λ, and after simple algebraic manipulation, we obtain

$$w_i = \frac{Q_{ii}^{-1}}{\sum_i Q_{ii}^{-1}}. \quad (4)$$

Here we used the fact that $Q_{ii} = p(x = \omega_1|R_i)p(x = \omega_2|R_i)$. Thus, a larger Q_{ii} implies a smaller weight. Since $p(x = \omega_1|R_i)+p(x = \omega_2|R_i) = 1$, we have $Q_{ii} = p(x = \omega_1|R_i)(1 - p(x = \omega_1|R_i))$. Therefore, when $p(x = \omega_1|R_i) = p(x = \omega_2|R_i) = \frac{1}{2}$, Q_{ii} will be the largest, which implies that w_i will be the smallest.

5 Estimation

The previous section shows that a large weight should be associated with a representation that provides better class separation, averaged over all sample variables x, in terms of the difference between $p(x = \omega_1|R_i)$ and $p(x = \omega_2|R_i)$. The larger the difference, the better the separation between two classes. There are several ways to estimate class separation from training data. In this paper, we focus on pooled variance ratio [4], which we show later is equivalent to linear discriminant analysis [5].

5.1 Pooled Variance Ratio

Variance ratio is proposed in [4] for determining weight w_i for the optimal combination of representations. Let p_1 represent the object distribution and p_2 represent the background distribution in representation R_i. Variance ratio can be defined as

$$VR(R_i; p_1, p_2) = \frac{var(R_i; (p_1 + p_2)/2)}{var(R_i; p_1) + var(R_i; p_2)}, \quad (5)$$

where $var(R_i; p)$ denotes the variance of R_i with respect to p. w_i can be computed as

$$w_i = VR(R_i; p_1, p_2)/\sum_j VR(R_j; p_1, p_2). \quad (6)$$

In [16], a theoretical justification for Eq. (6) is provided that shows that $w_i = c_{ii}^{-1}/\sum_j c_{jj}^{-1}$, where $c_{ii} = E[(p(x|R_i) - p(x|I))(p(x|R_i) - p(x|I))^t]$.

It states that w_i should be proportional to the accuracy of probability estimates provided by each representation. On the other hand, Eq. (6) says that w_i should be proportional to the degree of class separability. In general, one can make correct prediction as long as relative estimates are correct. On the other hand, our result provides a more direct foundation for the use of Eq. (6).

Let $S_t = \sum_i^n (x_i - m)(x_i - m)^t$, $S_b = \sum_{i=1}^2 n_i(m_i - m)(m_i - m)^t$, $S_i = \sum_{l(x_j)=i}(x_j - m_i)(x_j - m_i)^t$, where m denotes the overall mean, n_i represents the number of examples in class i, and $l(x)$ denotes the label of x. It is clear that $S_t = S_b + S_w$. Also, let $S_w = S_1 + S_2$. If we approximate $var(R_i; p_1, p_2)$ and $var(R_i; p_j)$ by $tr(S_t)$ and $tr(S_j)$ (within class scatter), respectively, we have $VR(L_i; p_1, p_2) = 1 + \frac{tr(S_b)}{tr(S_w)}$, If we ignore constant 1, the above shows that variance ratio (5) is identical to Fisher's LDA under appropriate conditions [7], in terms of computing linear discriminants. This allows us to compute w_i

$$w_i = VR_i/\sum_j^M VR_j. \quad (7)$$

5.2 Chernoff Distance

One way to measure class separation is the Chernoff distance [5]. If each class follows a Gaussian distribution, the Chernoff distance can be computed as the trace of matrix C [12]

$$C = S_w^{-1}(S_b - \tilde{S}_w) \quad (8)$$

where $\tilde{S}_w = S_w^{1/2}(p_1 \log(S_w^{-1/2} S_1 S_w^{-1/2}) + p_2 \log(S_w^{-1/} S_2 S_w^{-1/2}))S_w^{1/2})$.

Thus, for each representation R_i, we compute $Cher_{R_i} = tr(C_i)$. It follows that the optimal weight can be computed according to

$$w_i = Cher_{R_i}/\sum_j^M Chernoff_{R_j}. \quad (9)$$

6 Experiments

We have carried out empirical study evaluating the performance of the proposed algorithm. For simplicity, we experiment with decision trees (DTs) along each representation. As comparison, the following methods are evaluated: (1) variance ratio (Eq. 7), VR. (2) the Chernoff distance (Eq. 9), Cher.

Since tracking can be viewed as a classification problem, we use classification problems to validate our proposal. Three FERET image data sets and one gene data set are used. The problems are (1) Face detection, (2) Gender classification, and (3) detection of Glasses on faces. For the face and gender data, each image is represented by three poses in terms of eigenfaces extracted from three head orientations: 1) frontal, 2) half left, and 3) half right profiles. The non-face images are blacked

out faces. In the glass detection experiment, each image is represented by three types of features extracted from only one pose of an individual, namely (1) eigenfaces, (2) Canny edges, and (3) wavelet coefficients.

The gene data set is from the Yeast Database (CYGD) [13]. The task is to combine different sources to determine membrane vs non-membrane proteins. Three sources are derived from BLAST and Smith-Waterman genomic methods, and from gene expression measurement. The dataset has 100 examples and the number of dimensions after applying PCA is 76, 74 and 64, respectively. These dimensions explain 90% variance in the data.

Table 1. Average accuracy

Data	*Face*	*Gender*	*Glass*	*Gene*
VR	0.75	0.92	0.68	0.56
Che	0.76	0.93	0.68	0.54
DTs	0.74	0.83	0.59	0.49

The results are averaged over 30 runs (60% training and 40% testing). Table 1 shows the average accuracy. As a reference, the last row in Table 1 shows the best results that can be achieved by any representation. The fused methods performed better than the best single representation. Variance ratio and the Chernoff distance achieved similar performance, as expected.

7 Summary

We have introduced a novel technique for data fusion by minimizing KL divergence that measures the difference between an ideal (but unknown) probability and its estimate. We have shown that the resulting weight for a representation is related to the degree of class separability the representation can provide. We have provided a number of practical ways to estimate class separability. We have also provided experimental results that corroborate our theoretical analysis.

References

[1] S. Abney. Bootstrapping. In *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 360–367, 2002.

[2] A. Argyriou, A. Argyriou, M. Herbster, M. Herbster, M. Pontil, and M. Pontil. Combining graph laplacians for semi-supervised learning. In *Advances in Neural Information Processing Systems 18*, pages 67–74. MIT Press, 2005.

[3] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *In Proceedings of the Eleventh Annual Conference in Computational Learning Theory*.

[4] R. T. Collins, Y. Liu, and M. Leordeanu. On-line selection of discriminative tracking features. *IEEE Trans. on PAMI*, 27(10):1631–1643, 2005.

[5] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic Press, 1990.

[6] T. Joachims, N. Cristianini, and J. Shawe-Taylor. Composite kernels for hypertext categorisation. In *Proceedings of the 18th International conference on Machine learning*, pages 250–257. ACM Press, 2001.

[7] D. Juo, C. Ding, and H. Huang. Linear discriminant analysis: New formulations and overfit analysis. *Proceedings of the 25th National Conference on Artificial Intelligence (AAAI)*, pages 417–422, 2011.

[8] J. Kittler. Combining classifiers: A theoretical framework. *Pattern Analysis and Applications*, 1:18–27, 1998.

[9] S. Kullback and R. A. Leibler. On information and sufficiency. *Ann. Math. Statistics*, 22(11):79–86, 1951.

[10] L. I. Kuncheva, J. C. Bezdek, and R. P. W. Duin. Decision templates for multiple classifier fusion: An experimental comparison. *Pattern Recognition*, 34:299–314, 2001.

[11] G. R. G. Lanckriet, M. H. Deng, N. Cristianini, M. I. Jordan, and W. S. Noble. Kernel-based data fusion and its application to protein function prediction in yeast. *Proceedings of the Pacific Symposium on Biocomputing*, 9:300–311, 2004.

[12] M. Loog and P. Duin. Linear dimensionality reduction via a heteroscedastic extension of lda: The chernoff criterion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):732–739, 2004.

[13] H. W. Mewes, D. Frishman, C. Gruber, B. Geier, D. Haase, A. Kaps, K. Lemcke, G. Mannhaupt, F. Pfeiffer, C. Schller, S. Stocker, and B. Weil. Mips: a database for genomes and protein sequences. *Nucleic Acids Research*, 28(1):37–40, 2000.

[14] V. Sindhwani, P. Niyogi, and M. Belkin. A co-regularization approach to semi-supervised learning with multiple views. In *In Proceedings of the Workshop on Learning with Multiple Views, 22nd ICML*, 2005.

[15] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[16] Z. Yin, F. Porikli, and R. T. Collins. Likelihood map fusion for visual object tracking. In *IEEE workshop on Applications of Computer Vision*, pages 1–7, 2008.

[17] D. Zhang, F. Wang, C. Zhang, and T. Li. Multi-view local learning. *Proceedings of the 23rd National Conference on Artificial Intelligence (AAAI)*, pages 752–757, 2008.

[18] T. Zhang, A. Popescul, and B. Dom. Linear prediction models with graph regularization for web-page categorization. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 821–826. ACM Press, 2006.