Unsupervised Learning of Discriminative Edge Measures for Vehicle Matching between Non-Overlapping Cameras

Ying Shan Harpreet S. Sawhney Rakesh (Teddy) Kumar
Sarnoff Corporation
201 Washington Road
Princeton, NJ 08540
{yshan, hsawhney, rkumar}@sarnoff.com

Abstract

This paper proposes a novel method for matching road vehicles between two non-overlapping cameras. The matching problem is formulated as a same-different classification problem: probability of two observations from two distinct cameras being from the same vehicle or from different vehicles. We employ a novel measurement vector consists of three independent edge-based measures and their associated robust measures computed from a pair of aligned vehicle edge maps. The weight of each match measure in the final decision is determined by a novel unsupervised learning process so that the same-different classification can be optimally separated in the combined measurement space. The robustness of the match measures and the use of discriminant analysis in the classification ensures that the proposed method performs better than existing edge-based approaches, especially in the presence of missing/false edges caused by shadows and different illumination conditions, and systematic misalignment caused by different camera configurations. Extensive experiments based on real data of over 200 vehicles at different times of day demonstrate promising results.

1 Introduction

Figure 1: Vehicle images at different cameras along a road network, where the cameras are roughly oriented perpendicular to the direction of motion along the road. The top row shows vehicle images in the near lane, and the bottom row shows those in the far lane

This paper addresses the problem of matching vehicles as they are imaged in non-overlapping fixed cameras along a road network. The cameras are roughly oriented perpendicular to the direction of motion along the road. Figure 1 shows images of the same vehicle from multiple cameras to demonstrate a variety of changes in appearance and pose. For the problem of maintaining a track of vehicles over multiple cameras, a key component is feature-based vehicle matching between observations from a pair of cameras. In addition to the cameras being physically different with different optical and internal geometric characteristics, the temporal and spatial separation between two observations from two cameras involves changes due to pose of vehicles, illumination conditions and the position of shadows due to environmental structures such as trees, light poles and buildings. Our vehicle matching approach combines approximate knowledge of the relative geometry of vehicles and cameras, with robust match measures and discriminant analysis to compute the probability of two observations being of the same vehicle versus different vehicles. The approach relies on edge feature extraction, discriminant-based combination of robust match measures and unsupervised learning.

We pose the vehicle matching problem as a two class classification problem, where a discriminative match score is computed by combining multiple edge-based measures through unsupervised learning. The classifier produces the probability of the score between observations from two cameras given that the observations belong to the same vehicle, and the probability of the score given that the observations belong to different vehicles. Given the two-class probabilities, a global tracker that maintains multiple hypotheses for vehicles across multiple cameras can make optimal decisions based on multi-hypotheses filters such as the Joint Probability Density Association Filter (JPDAF) [1] or Probabilistic Multi-Hypotheses Tracker (PMHT) [2]. Note that a detailed discussion of the incorporation of these probabilities into a tracker is beyond the scope of this paper.

A key step involves learning of same/different probability density functions. We compute a novel multi-dimensional measurement vector and demonstrate its use in learning these distributions both in a supervised and unsupervised modes. In the supervised mode, a set of training data with same/different labels is used to compute the maximally discriminative distributions using Fisher's Linear Discriminants. We then extend the discriminative learning of distributions to the unsupervised case using a sampling algorithm that efficiently exploits the space of the weights and finds the optimal solution combining the edge measures. This algorithm achieves almost the same correct classification performance in the unsupervised case as for the supervised case.

The unsupervised discriminative learning framework proposed in this paper is independent of any specific features. However, edge-based measures are chosen since edge

features are the dominant features in vehicular objects and they remain relatively stable over aspect and illumination variations. Each edge map is computed from the masked area of a vehicle chip, where the masks are provided by a real-time tracker running on each camera in the road network.

2 Related Work

The technical components of the proposed work are related to the previous work on edge-based object matching, and learning robust and discriminative measures for classification. Object matching with edge features has been proved to be reliable in previous work. In [3, 4, 5], edge features were used to detect traffic signs, pedestrians, and for recognizing hand gestures. Examples of traditional edge-based match measures include Chamfer distance [3], Hausdorff distance [6], and Earth Mover's distance [7]. In [8, 5], both edge locations and orientations are used to define a combined edge measure, which is reported to improve the matching and classification performance significantly. The SIFT descriptor introduced by Lowe in [9] uses aggregated measure computed from both gradient orientation and magnitude to tolerate slight location errors.

There are two issues related to edge-based measures: robustness and feature selection/combination. Many previous works use clean edge maps for at least one of the edge maps. Truncated Chamfer distance [6] or robust Hausdorff distance [8] may work for these cases, but not for the cases when both edge maps are not clean.

The issue of selection and combination of discriminative edge measures to maximize the overall classification performance is a main focus of this paper as also in [10, 11]. Wu et. al. [10] address the problem of learning discriminative image features with a limited set of labeled data based on a semi-supervised learning approach. Collins et. al. [11] address the problem of on-line selection of discriminative color features for tracking. The learning is based on a set of foreground pixels and background pixels labeled by the tracker with a "center-surround" approach. It is possible that the result can be biased by pixels that are incorrectly labeled. In contrast with these methods, our approach is unsupervised, and does not involve a fixed label set.

On the application side, [12] also deals with object matching between non-overlapping cameras and on-line learning of camera topology and path probabilities, but the focus is not on selecting or combining measures. The work in [13] and [14] propose a nice framework for object matching and feature learning, but they use only simple measures such as color and size.

3 Problem Statement and Notations

For a given pair of cameras C_i and C_j, we want to estimate the probability density functions:

$$\begin{array}{c} P(y \mid \text{same}, C_i, C_j) \equiv P(y \mid \mathcal{S}_{i,j}) \\ P(y \mid \text{different}, C_i, C_j) \equiv P(y \mid \mathcal{D}_{i,j}) \ , \end{array} \tag{1}$$

where $P(y \mid \mathcal{D}_{i,j})$ and $P(y \mid \mathcal{S}_{i,j})$, are the probability density functions of the *match score* y given that the two observations are of same/different vehicles, and

$$y = f_{i,j}(E_k^i, E_l^j) \ , \tag{2}$$

where y is a scalar function of two observed edge maps, E_k^i and E_l^j, corresponding to the kth and lth observations in cameras C_i and C_j, respectively. The two edge maps are assumed to be approximately aligned typically using parametric alignment models and ICP algorithm, but both could be contaminated by noise, scene clutter, and obscuration. Each edge map is also time stamped. The problem is, without manually labeling the data, how to design the function $f_{i,j}$ so that $P(y \mid \mathcal{D}_{i,j})$ and $P(y \mid \mathcal{S}_{i,j})$ are maximally separated.

4 Learning Robust Edge Measures

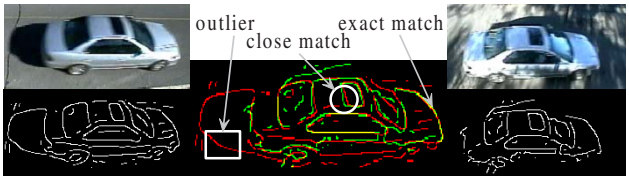

Figure 2: An example of edge map alignment (middle image) of two vehicle images (top left and top right), and their corresponding edge maps (bottom left and bottom right). This figure is best viewed with color

Figure 2 shows a typical alignment result of two edge maps and their corresponding images. Yellow pixels in the figure are perfect matches. Edge pixels such as those shown in the circle are approximate matches. Other pixels such as those in the square are outliers that do not have any close matches. We exploit the information content in the edge maps by computing a six dimensional measurement vector. Three of the components measure spatial, orientation and magnitude differences between matching features. Separation of matching features into inliers and outliers provides us with another set of useful match measures: the coverage of features for each match measure. Each set of inliers for a match measure gives the percentage of the features included in the match measure. Ideally, matching observations should not only have low distances corresponding to the three matching features but for each feature the coverage should be high indicating that a high degree of correlation is present between the two edge maps. Thus, each of the three match measures is augmented with its corresponding coverage measure. The optimal combination of the component match measures can then be determined by a process described in Sec. 5.

4.1 Raw Edge Measures

The six-dimensional match measure is derived from three pixel-to-pixel measures as shown in Fig. 3. Suppose that $\mathcal{M}$ and $\mathcal{I}$ are two aligned edge point sets, p is a point in $\mathcal{M}$, and q is the closest point of p in $\mathcal{I}$, we define

$$d^p_{\mathcal{M} \mapsto \mathcal{I}} = \|p - q\|_1 \ , \tag{3}$$

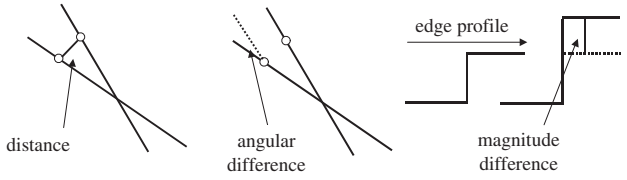

Figure 3: Raw measures: pointwise distance, angular difference, and gradient magnitude difference

$$a^p_{\mathcal{M}\mapsto\mathcal{I}} = \theta_p - \theta_q \ , \tag{4}$$

$$m^p_{\mathcal{M}\mapsto\mathcal{I}} = \text{mag}_p - \text{mag}_q \ , \tag{5}$$

where d, a, and m denote distance, angular difference, and magnitude difference, respectively; $\theta_{\{p,q\}}$ and $mag_{\{p,q\}}$ are the edge directions, and gradient magnitudes defined on the edge points p and q, respectively. The subscript $\mathcal{M} \mapsto \mathcal{I}$ denotes that the closest point is defined from $\mathcal{M}$ to $\mathcal{I}$.

4.2 Derived Edge Measure

Based on the three raw measures, the distance measure between a pair of edge maps is derived as:

$$\mathbf{v}_{\mathcal{M}\leftrightarrow\mathcal{I}} = [\tilde{d}, c^d, \tilde{a}, c^a, \tilde{m}, c^m] \ , \tag{6}$$

where the subscript $\mathcal{M} \leftrightarrow \mathcal{I}$ denotes that the measure is defined symmetrically. The first measure $\tilde{d} \equiv \tilde{d}_{\mathcal{M}\leftrightarrow\mathcal{I}}$ is the average inlier distance defined as:

$$\tilde{d} = \frac{\sum_{p\in\mathcal{G}^d_{\mathcal{M}\mapsto\mathcal{I}}} d^p + \sum_{p\in\mathcal{G}^d_{\mathcal{I}\mapsto\mathcal{M}}} d^p}{N(\mathcal{M}) + N(\mathcal{I})} \ , \tag{7}$$

where $\mathcal{G}^d_{\mathcal{X}\mapsto\mathcal{Y}}$ is the set of points in $\mathcal{X}$ corresponding to the inlier distances defined from $\mathcal{X}$ to $\mathcal{Y}$, and $N(\mathcal{X})$ is the number of total edge points in $\mathcal{X}$. Whether a distance measure is an inlier or an outlier is determined by the approach discussed in Sec. 4.3. The second dimension $c^d \equiv c^d_{\mathcal{M}\leftrightarrow\mathcal{I}}$ is the cardinality of the set of points covered by the inlier distances, and is defined as:

$$c^d = \frac{N(\mathcal{G}^d_{\mathcal{M}\mapsto\mathcal{I}}) + N(\mathcal{G}^d_{\mathcal{I}\mapsto\mathcal{M}})}{N(\mathcal{M}) + N(\mathcal{I})} \ . \tag{8}$$

The other four measures related to the raw angular difference a, and magnitude difference m are defined in a similar way.

4.3 Estimating Inlier and Outlier Distributions

A raw measure x^p, $x \in \{d, a, m\}$, at a point p can be classified as an inlier or an ourlier as follows:

$$x^p \in \begin{cases} \mathcal{G}^x, \text{ if } B_x(x^p) < G_x(x^p) \\ \mathcal{B}^x, \text{ otherwise} \end{cases} \ , \tag{9}$$

where $\mathcal{B}^x$ is the set of outliers for the raw measure x, B_x the probability density function of ourliers, and G_x is the density function of inliers. Assuming that both B_x and G_x are Gaussian, the parameters of these distributions can be computed by collecting a set of aligned edge maps $\{\mathcal{M}_i \leftrightarrow \mathcal{I}_i\}$, and computing raw measures $x^p_{\mathcal{M}_i\mapsto\mathcal{I}_i}$ and $x^p_{\mathcal{I}_i\mapsto\mathcal{M}_i}$ for all the pixels in each pair. The standard EM algorithm is then applied to the set of all the raw measures to estimate the parameters of a Gaussian mixture model with two components. Since the outlier distribution should be close to a uniform distribution, the inlier distribution is the component with the smaller σ. We then have $B_x = \mathcal{N}_x(\mu^x_b, \sigma^x_b)$, and $G_x = \mathcal{N}_x(\mu^x_g, \sigma^x_g)$, where $\mathcal{N}$ denotes a normal distribution, and $\sigma^x_g < \sigma^x_b$. Fig. 4 shows the inlier and outlier distributions estimated from two pairs of cameras for the poinwise distance, angular difference and magnitude difference match measures. Table 1 lists the parameters of inlier distributions of three raw measures for 8 pairs of cameras. Note that the camera indices are from 1 to 16 for 8 cameras. Odd numbered and even numbered indices refer to vehicles traveling in the near and far lane, respectively.

To ensure the success of the EM algorithm, each component needs to have enough samples. For the results presented in Fig. 4 and Table 1, the set $\{\mathcal{M}_i \leftrightarrow \mathcal{I}_i\}$ contains only those pairs of edge maps that correspond to same vehicles across the two cameras. In Sec. 6 we will show how to compute B_x and G_x without knowing the ground truth. Note that the inlier distribution for d is more like a Rayleigh distribution.

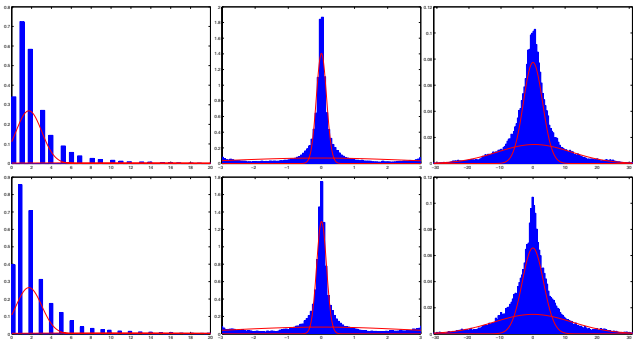

Figure 4: The estimated inlier and outlier distributions overlaid with the corresponding histograms for 2 different pairs of cameras. Left: Pointwise distance, Middle: Angular difference, Right: Magnitude distance. This figure is best viewed with color

5 Learning Discriminant Edge Measures

Given the six-dimensional distance measure described in Sec. 4 the problem now is to determine the weight of each individual component in the distance measure such that $P(y|\mathcal{D})$ and $P(y|\mathcal{S})$ of the combined match score y are maximally separated. If y is assumed to be a linear combination of the components of $\mathbf{v}$, this problem can be solved with the Fisher's Linear Discriminants (FLD). Given two sets of n six-dimensional samples $\{\mathbf{v}_i | i = 0, \cdots, n-1\}$ collected from two distinct cameras, n_1 in the set $\mathcal{V}_1$ labeled as true matches, and n_2 in the set $\mathcal{V}_2$ labeled as wrong matches. If we project each vector onto a line in the direc-

C_i	C_j	μ_d	σ_d	μ_a	σ_a	μ_m	σ_m
3	1	1.73	1.30	0.002	0.15	-0.02	2.98
7	3	1.73	1.29	0.005	0.15	-0.06	3.31
2	4	1.19	0.89	-0.001	0.17	-0.11	3.42
11	7	1.40	1.06	0.001	0.13	-0.06	2.96
4	8	1.11	0.87	0.000	0.17	-0.15	4.34
15	11	1.72	1.27	-0.002	0.15	-0.21	3.14
8	12	1.30	0.93	-0.002	0.14	-0.14	3.74
12	16	1.44	1.04	0.007	0.14	-0.10	4.78

Table 1: Parameters of inlier distributions for 8 pairs of cameras. The left two columns are the camera indices. From left to right of the rest columns are the μ and σ for the three raw measures

tion of $\mathbf{w}$ such that

$$y_i = \mathbf{w}^t \mathbf{v}_i \ , \tag{10}$$

the original n samples are then divided into the corresponding subset $\mathcal{Y}_1$ of true matches and $\mathcal{Y}_2$ of wrong matches. The best line direction $\mathbf{w}$, i.e., the weight vector, can be found by maximizing the criterion function:

$$\mathbf{J}(\mathbf{w}) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{s}_1^2 + \tilde{s}_2^2} \ , \tag{11}$$

where $\tilde{\mu}_{\{1,2\}}$ are the sample means, and $\tilde{s}^2_{\{1,2\}}$ are the scatters for the projected sample sets $\mathcal{Y}_{\{1,2\}}$. Once the optimal $\mathbf{w}$ is obtained, we have:

$$\begin{array}{l} P(y|\mathcal{S}) = \mathcal{N}(\tilde{\mu}_1, \tilde{s}_1^2/n_1) \\ P(y|\mathcal{D}) = \mathcal{N}(\tilde{\mu}_2, \tilde{s}_2^2/n_2) \ , \end{array} \tag{12}$$

where $\mathcal{N}$ denotes a normal distribution.

Table 2 shows the weights computed for 8 pairs of cameras using this approach. The weights are normalized with the μ and σ of each measure for a meaningful comparison. It can be observed that in most cases, the angular difference has high weight. This is consistent with the observation in [8, 5] that adding edge orientation leads to significantly better classification results. It should also be noted that the coverage features such as c^a and c^m are also the measures with large weights. Therefore, a combination of the distance features and the amount of data explained by the inliers is an effective feature set for the problem at hand. A nice property of this approach is that it automatically selects measures that are important for the particular pair of cameras under consideration. For the pair of 15 and 11, the angular measure seems to be most discriminative since the magnitude change between these two cameras is the largest among all the pairs of cameras (see Table 1 for the distribution of each measure). In the case of 11 and 7, the magnitude measure plays an important role, since the magnitude change is the smallest among all the pairs of cameras.

6 Unsupervised Learning of Weights and Distributions

The approach proposed in Sections 4 and 5 for learning robust and discriminative edge measures requires a large amount of labeled data, which is difficult to obtain for a system with many cameras. Furthermore, distributions computed at one time of the day may not be suitable for the situation at another time or day. To address these problems, we propose an unsupervised approach in which robust and discriminative edge measures can be learned without labeled data. The algorithm is designed to be executed in a batch mode during run time. More specifically, the on-line system keeps collecting vehicle images for a certain period of time, say 20 to 30 minutes. The estimation of discriminative distributions is based on the latest data so that the distributions can always be adapted to the current situation. As a result, this approach is able to re-initialize itself without any manual input. The approach can also be modified to run in a continuous manner using the distributions that were already computed at an earlier time. The algorithm is outlined here with details to follow below:

Measures		$\tilde{d}$	c^d	$\tilde{a}$	c^a	$\tilde{m}$	c^m
Weights		w_0	w_1	w_2	w_3	w_4	w_5
3	1	0.05	0.02	0.14	0.05	-0.12	0.07
7	3	0.08	0.03	0.12	0.18	0.03	0.06
2	4	0.04	-0.01	-0.13	0.18	-0.07	0.10
11	7	0.01	0.00	-0.01	0.12	0.32	0.27
4	8	0.05	-0.01	0.14	0.13	-0.04	0.08
15	11	0.01	0.00	-0.15	0.03	0.06	0.04
8	12	-0.17	-0.07	0.13	0.05	0.00	0.08
12	16	-0.08	0.24	0.05	0.26	-0.14	0.09

Table 2: Weights of the corresponding measures computed for 8 pairs of cameras

1. Collect an unlabeled set $\mathcal{E}$ of aligned pairs of edge maps, and ensure that there is sufficient amount of true matches in the sample set.
2. Learn outlier and inlier distributions based on $\mathcal{E}$.
3. Compute the set $\mathcal{V}$ of six-dimensional derived measures based on $\mathcal{E}$ and the outlier and inlier distributions learned in Step 2, and learn the weight $\mathbf{w}$, $P(y|\mathcal{D})$, and $P(y|\mathcal{S})$ from $\mathcal{V}$ simultaneously.

6.1 Ranking Vehicle Matches for Sample Collection

The major challenge of building a representative sample set for unsupervised learning is to collect sufficient percentage of true matches in the set. We address this problem using a ranking mechanism with *time gating* as a pre-pruning stage. Let us consider traffic flow between two cameras, C_i to C_j, and recall that each vehicle image has a time stamp. A vehicle in C_j is said to be time gated with a vehicle in C_i only if its transition time from C_i to C_j is within a certain range of the average transition time between these two cameras. For each vehicle edge map E_k^j in C_j, we form a candidate list $\mathcal{C}_k^j$ of edge maps of all the vehicles in C_i that time gated with E_k^j. For each edge map in $\mathcal{C}_k^j$, we compute a ranking score, and then sort the candidate list from high to low according to the score. For each vehicle edge map E_k^j, we then select K samples with top scores in its candidate list, and L samples in the rest of the list and call it the sample

set $\mathcal{E}$. The ranking score is defined as:

$$\gamma = \frac{\sum_{\mathcal{M}\mapsto\mathcal{I}} h(d^p,\delta)h(a^p,\alpha) + \sum_{\mathcal{I}\mapsto\mathcal{M}} h(d^p,\delta)h(a^p,\alpha)}{N(\mathcal{M}) + N(\mathcal{I})} \tag{13}$$

where $\mathcal{M}$ and $\mathcal{I}$ are the two edge point sets, $h(x,c) = (1-|x|/c)$ for $|x| < c$, and $h(x,c) = \rho$ for $|x| \geq c$, and ρ is a small positive number; d^p and a^p are as in (3) and (4). The constants δ and α are kept the same for all pairs of cameras. The score in (13) is in the range of $[0,1]$. The score converts the pointwise distance and angular difference, and their coverages in terms of inliers and outliers into a single linear inverted hat like robust match measure. Edge magnitude differences are not used for ranking since they are relatively more sensitive to illumination changes. The ranking score is not as discriminative as the discriminative match score computed in (10) using the six-dimensional edge measures. However, it is sufficient to serve the purpose of relative ordering amongst candidate matches that helps in choosing training data for learning.

Table 3 shows the ranking results computed for 8 pairs of cameras, with an average of 190 vehicles between each pair. The "Total" column lists the number of total vehicles with ground truthed matches, and the "Top 2" column lists the number of true matches that are ranked within the top 2. The table also shows that the average probability that the true matches are ranked within the top 2 is about 0.96. In other words, if we collect 2 top ranked samples and 1 sample from the rest of the candidate list for each vehicle, on the average $95.56/3 = 31.9\%$ of matches will be true matches in the resulting sample set $\mathcal{E}$. This is sufficient for unsupervised learning.

C_i	C_j	Total	Top 2	Percentage
3	1	195	186	95.38
7	3	197	193	97.97
2	4	152	148	97.37
11	7	228	220	96.49
4	8	166	165	99.4
15	11	210	193	91.9
8	12	185	172	92.97
12	16	186	173	93.01

Table 3: Ranking of matches for vehicles traveling from C_i to C_j. The average probability that the true matches are ranked within the top 2 is 0.9556

6.2 Estimating Outlier and Inlier Distributions

A two component Gaussian mixture model is fitted to scores in the set $\mathcal{E}$ obtained from the previous steps. Each component is essentially the $P(\gamma|\mathcal{D})$ and $P(\gamma|\mathcal{S})$ for the score. The component with the larger mean accounts for the true matches. Since the score is not optimally designed for separating wrong matches from true matches, we conservatively pick those samples as inliers for which $P(\gamma|\mathcal{D}) \ll P(\gamma|\mathcal{S})$. Note that at this stage, the samples are not labeled as true and false matches; the fitted mixture model is used only to prune the set of scores and the corresponding matches.

6.3 Learning Weights and Distributions

Once the outlier and inlier distributions are known, we can compute a set $\mathcal{V} = \{\mathbf{v}_0, \mathbf{v}_1, \cdots, \mathbf{v}_{m-1}\}$ of derived measurement vectors from the sample set $\mathcal{E}$ as described in Sec. 4.2. Now the goal is to compute the weight, $\mathbf{w}$, as in (10), along with the discriminative distributions as in (12) without explicit labeling of the samples.

We first discretize $\mathbf{w} \in \Omega \equiv \{l_0, l_1, \cdots, l_{n-1}\}^6$, where n is the number of discrete samples for each of the six dimensions, and l_i are the values uniformly distributed in the range of $[-1,1]$ under the assumption that each of the six measures can be normalized within its range. The most discriminative weight $\mathbf{w}^*$ can then be obtained by solving the following optimization problem:

$$\max_{\mathbf{w}\in\Omega} \mathbf{J}(\mathbf{w}) \ , \tag{14}$$

where the objective function is defined as

$$\frac{|\mu_1 - \mu_2|^2}{\sigma_1^2 + \sigma_2^2} \ . \tag{15}$$

Given any $\mathbf{w}$, $\mu_{\{1,2\}}$ and $\sigma_{\{1,2\}}$ are computed by first projecting the set of $\mathcal{V}$ onto the direction $\mathbf{w}$, and then fitting a two component Gaussian mixture model on the projected samples. In other words, for any hypothesized direction $\mathbf{w}$, the unlabeled samples are described using a two-component mixture model, and that model that produces the maximal discrimination chosen as the final solution.

Solving (14) with exhaustive search requires 6^n operations, which is tractable only when n is small. When n is large, a Gibbs sampler [15] is employed to explore the discrete solution space efficiently. The Gibbs distribution that corresponds to the objective function in (14) is given by:

$$G(\mathbf{w}) = \frac{1}{Z}\exp[\mathbf{J}(\mathbf{w})/T] \ , \tag{16}$$

where Z is an unknown normalization factor, and T is the temperature constant. The local conditional probability can be derived easily from (16) as:

$$\begin{aligned} & G(w_j = l_k \,|\, \{w_i | i \neq j\}) \\ = \ & \frac{G(w_j = l_k, \{w_i\})}{G(\{w_i\})} \\ = \ & \frac{G(w_j = l_k, \{w_i\})}{\sum_{w_j \in \{l_k | k=0,\cdots,n-1\}} G(w_j, \{w_i\})} \ , \end{aligned} \tag{17}$$

where w_j is the jth dimension of $\mathbf{w}$. Note that the unknown factor Z is canceled out in (17). In order to compute $G(w_j, \{w_i\}) \equiv G(\mathbf{w}^j)$ for all $\{\mathbf{w}_k^j | w_j = l_k, k = 0, \cdots, n-1\}$, we first write the projection $\mathbf{Y}^k \equiv [y_0^k, \cdots, y_{m-1}^k] = \mathbf{w_k^j}^T [\mathbf{v}_0, \cdots, \mathbf{v}_{m-1}]$ of the sample set $\mathcal{V}$

onto each $\mathbf{w}_k^j$ as the following matrix form

$$[w_0, \cdots, w_j = l_k, \cdots, w_5] \begin{bmatrix} v_0^0 & \cdots & v_1^0 & \cdots & v_{m-1}^0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_0^j & \cdots & v_1^j & \cdots & v_{m-1}^j \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_0^5 & \cdots & v_1^5 & \cdots & v_{m-1}^5 \end{bmatrix}$$
$$\equiv [\mathbf{w}_{0,j-1}^T, w_j = l_k, \mathbf{w}_{j+1,5}^T] \begin{bmatrix} \mathbf{V}^{0,j-1} \\ \mathbf{v}^j \\ \mathbf{V}^{j+1,5} \end{bmatrix} . \tag{18}$$

It is then easy to see from (18) the following recursive formulas to efficiently compute $\mathbf{Y}^k$

$$\begin{aligned} \mathbf{Y}^0 &= \mathbf{w}_{0,j-1}^T \mathbf{V}^{0,j-1} + \mathbf{w}_{j+1,m-1}^T \mathbf{V}^{j+1,5} \\ \mathbf{Y}^{k+1} &= \mathbf{Y}^k + \eta\, \mathbf{v}^j \ , \end{aligned} \tag{19}$$

where $\eta = l_k - l_{k-1}$ the step between two consecutive levels. From $\mathbf{Y}^k$ the μ and σ of the Gaussian mixture model and hence $G(\mathbf{w}^j)$ can all be computed. The mixture model computed from $\mathbf{Y}^k$ can be used to initialize the EM algorithm for $\mathbf{Y}^{k+1}$ to save computational time.

Given a random initial guess, the sampler sweeps through each dimension w_j sequentially, and assigns its value according to the local conditional probability in (17). The same process is repeated for several iterations, and the $\mathbf{w}$ that has the smallest $G(\mathbf{w})$ is selected as the solution. Because our objective function is simple, and the dimension is relatively small, Gibbs sampler can quickly converge to a solution very close to the global optimum.

To prevent the singular case that only a few samples are assigned to one component of the mixture model, we enforce in the sampling process the constraint that the prior for each component also estimated from the EM algorithm should be larger than a threshold. This threshold is related to the percentage of the true matches in the sample set, which is around 31.9% according to the discussions in Sec. 6.1. In practice, we found 0.2 is a threshold good for all pairs of cameras.

7 Experiments

Figure 5: Vehicle examples in the databases

We collected two databases of vehicle chips from 8 pairs of cameras for the experiments. Figure 5 shows some vehicle examples in the databases. Both databases contain about 200 different vehicles passing through the cameras within a 30 minute time period. The first database, denoted as *DBM*, was collected around 10 am on one day. The second, denoted as *DBF*, was collected around noon on another day. We manually truthed around 1000 matching pairs of vehicles for *DBM*, and 1500 pairs for *DBF*.

Three major results are given in this section. First, we demonstrate that the match score computed with supervised learning approach is more robust and discriminative when compared with the existing edge-based measures. Second, we show the error rates of the classification results using the distributions trained using the supervised learning approach. Finally, we show that the unsupervised learning approach has performance comparable to the supervised learning approach. All the algorithms are tested with both databases. However, most results presented are from DBF because of the page limitation. In all the experiments, we use half of the truth samples for learning and the other half for testing.

7.1 Comparison with Other Edge-Based Measures

We have compared the discriminative match score proposed in Sec.4.2 with three other representative edge-based measures, i.e., Hausdorff distance [6], Truncated Chamfer distance[6]. We also tried to compare with the combined edge distance and orientation measure introduced in [8], but it didn't work well because of the larger percentage of outliers in our test sets. Instead, we compare with the ranking score defined in (13) since it also consists of both edge distance and orientation, but is not sensitive to outliers. The constant δ in (13) is set to be 5 pixels, and α is set to 15 degrees. The weight $\mathbf{w}$ is computed using the supervised learning approach discussed in Sec. 5. We use all the positive samples from the *DBF* database, and negative samples are randomly sampled from the same database. Based on the computed distributions of positive and negative samples, we plot the ROC curve for each approach, as has been shown in Fig. 6. It can be seen that performance varies at different pairs of cameras. However, in all cases, the proposed approach has always the highest performance. It is also interesting to note that the ranking score performs better than the two traditional approaches.

7.2 Classification Results

We use the $P(y|\mathcal{D})$ and $P(y|\mathcal{S})$ computed from the supervised learning approach discussed in Sec. 5 as the classifiers, and report error rates of 8 pairs of cameras for both *DBM* and *DBF* databases in Table 4. Despite that DBF has more shadows than *DBM*, the performance of our method holds quite well.

7.3 Comparison of Unsupervised Learning and Supervised Learning

We have compared the unsupervised learning approach introduced in Sec. 6 and the supervised approach discussed in Sec. 5. We use the same database *DBF* as in the first experiment. The sample set $\mathcal{E}$ is collected using the method

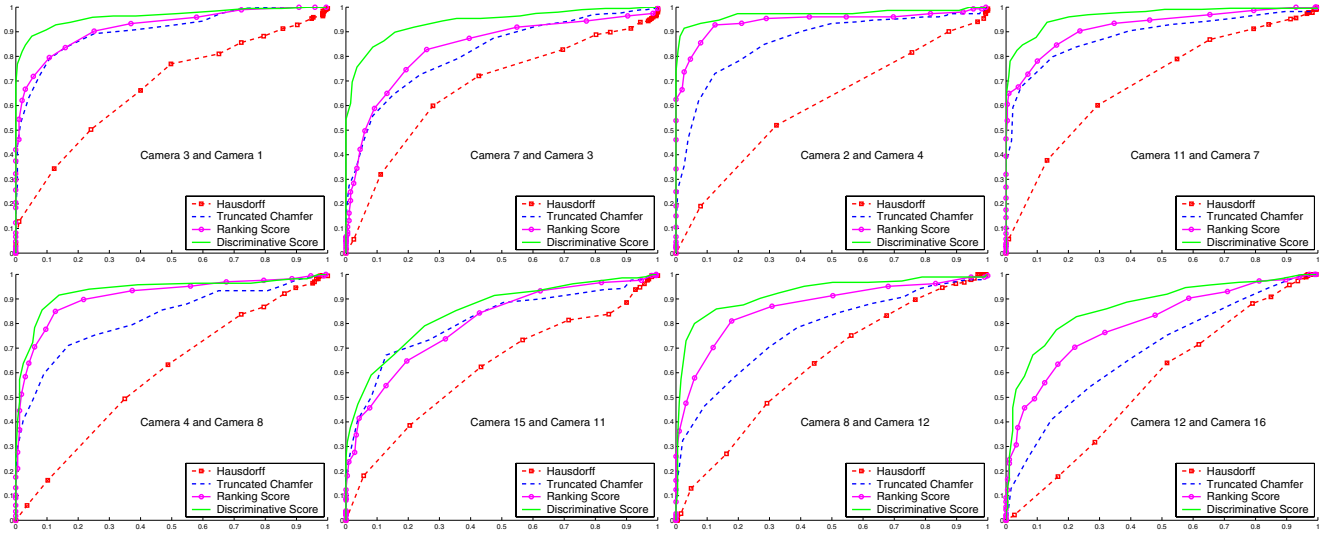

Figure 6: The results of the discriminative match score as comparison with other approaches. Weights are computed with the supervised learning approach discussed in Sec. 5. Each ROC plot depicts the probability of correct matches on the y-axis versus the probability of false matches on the x-axis

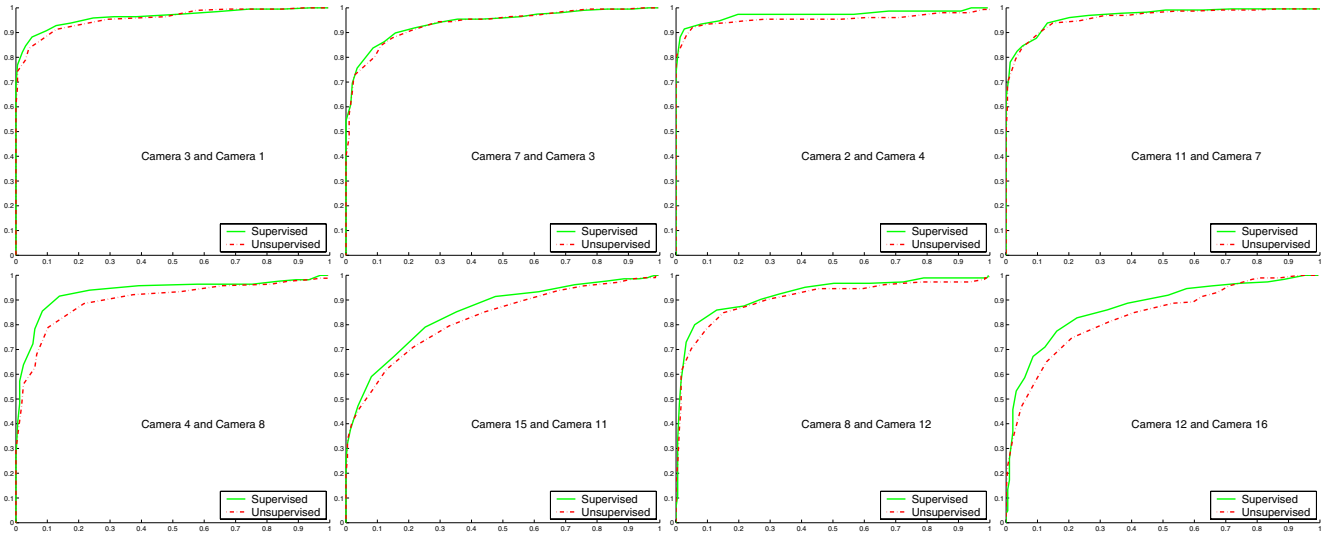

Figure 7: The results of the unsupervised learning approach as compared with the supervised learning approach. The ROC curves are plotted in the same way as in Fig. 6

3,1	7,3	2,4	11,7	4,8	15,11	8,12	12,16
0.09	0.13	0.08	0.11	0.13	0.25	0.16	0.19
0.08	0.19	0.09	0.12	0.13	0.28	0.19	0.25

Table 4: Error rates computed from different training and testing databases. The first row is the indices of the camera pairs, the second row is the error rates for database *DBM*, and the third row for *DBF*. The average error rates are 0.107 and 0.142 for *DBM* and *DBF*, respectively

discussed in Sec. 6.1, and K and L are set to be 2 and 1, respectively. We use $n = 20$ levels for each dimension in the weight $\mathbf{w}$, and set $T = 0.5$ in the Gibbs sampler. The best $\mathbf{w}^*$ is selected from a set of $\mathbf{w}$ after the sampler converges. Figure 7 shows the results of 8 pairs of cameras. It can be seen that the performance of the unsupervised learning approach is always slightly lower than the supervised method, but is in general much better than any of other measures shown in Fig. 6. Figure 8 shows the convergence of the Gibbs sampler for 3 pairs of cameras. It can be seen that Gibbs sampler usually converges within 50 iterations. Figure 9 is the estimated $P\{y|\mathcal{D}\}$ and $P\{y|\mathcal{S}\}$ of

3 pairs of cameras overlaid with corresponding normalized histograms of y projected from the sample set $\mathcal{V}$. It clearly shows that the projected measure y indeed forms Gaussian-like distributions for both wrong matches and true matches.

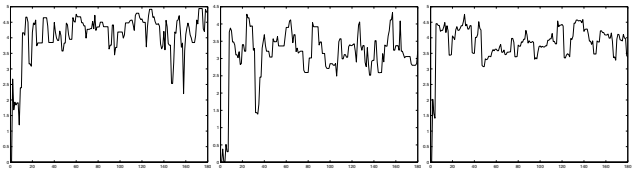

Figure 8: From left to right, the convergence of the Gibbs sampler for the camera pairs (3,1), (4,8), and (12,16), respectively. The horizontal axis is the iteration number, and the vertical axis is the $J(\mathbf{w})$ value defined in (14)

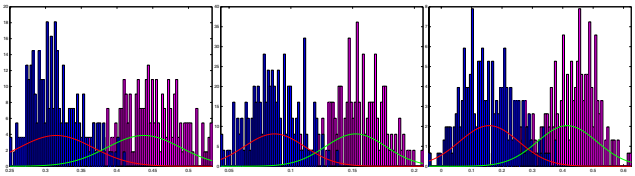

Figure 9: From left to right, the histograms and the $P(y|\mathcal{D})$ and the $P(y|\mathcal{S})$ distributions of the camera pairs (3,1), (2.4), and (11,7), respectively. The histograms of true matches and wrong matches are displayed in magenta and blue, respectively. The estimated two Gaussian components are displayed in green for true matches and red for wrong matches, respectively. This figure is best viewed with color

8 Conclusion and Future Work

We have proposed an unsupervised learning algorithm to compute robust and discriminative edge measures for matching vehicles between non-overlapping cameras. We have demonstrated the need to separate edge measures into outlier and inlier distributions, and use inlier coverages as additional measures. We have verified the power of discriminant learning when combining the new set of edge measures into a single match score. We have also designed a new algorithm for unsupervised discriminant learning without explicit and implicit labeling.

A key assumption in our unsupervised learning algorithm is that the projection of our edge measure onto a line can be fitted by a Gaussian mixture of two components. This has to do with the fact that the edge maps of different vehicles (negative samples) can not be arbitrary different. An interesting future direction is to modify this approach to deal with the problem of matching both vehicle and people between non-overlapping cameras. This is a more challenging problem because the distribution of negative samples can no longer be modeled with a single Gaussian distribution.

Acknowledgements

This work was performed under a DARPA Contract No. NBCHC030085. Distribution is unlimited for public release. This research is focused on vehicle tracking applications in restricted environments such as military bases and similar restricted access sites. All imagery used in this research is from restricted access environments. Any representations of civilians or civilian vehicles are an artifact of the experimental process and are used purely to expedite the research.

References

[1] B. Zhou and N.K. Bose. Multitarget tracking in clutter: fast algorithms for data association. *IEEE Trans. Aerospace and Electronic Systems*, 29(2):352–363, 1993.

[2] Cox I.J. and Hingorani S.L. An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 18(2):138–150, 1996.

[3] D.M. Gavrila and V. Philomin. Real-time object detection for "smart" vehicles. In *International Conference on Computer Vision (ICCV99)*, pages 87–93, 1999.

[4] Vassilis Athitsos and Stan Sclaroff. Estimating 3D hand pose from a cluttered image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR04)*, 2003.

[5] B. Stenger, A. Thayananthan, P. Torr, and R. Cipolla. Hand pose estimation using hierarchical detection. In *International Workshop on Human-Computer Interaction, Lecture Notes in Computer Science, vol. 3058, Springer (2004)*, 2004.

[6] D. Huttenlocher, G. Klanderman, and W.J. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 9(15):850–863, 1993.

[7] K. Grauman and T. Darrell. Fast contour matching using approximate Earth Mover's Distance. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR04)*, volume I, pages 220–227, 2004.

[8] Clark F. Olson and Daniel P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Trans. Image Processing*, 6(1):103–113, 1997.

[9] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[10] Ying W., Qi T., and T. S. Huang. Discriminant-EM algorithm with application to image retrieval. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR00)*, volume I, pages 222–227, 2000.

[11] R. Collins and Y. Liu. On-line selection of discriminative tracking features. In *International Conference on Computer Vision (ICCV03)*, 2003.

[12] O. Javed, Z. Rasheed, K. Shafique, and M. Shah. Tracking across multiple cameras with disjoint views. In *International Conference on Computer Vision (ICCV03)*, volume 2, pages 952–957, 2003.

[13] V. Kettnaker and R. Zabih. Bayesian multi-camera surveillance. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR99)*, 1999.

[14] Hanna Pasula, Stuart J. Russell, Michael Ostland, and Yaacov Ritov. Tracking many objects with many sensors. In *International Joint Conferences on Artificial Intelligence (IJCAI99)*, pages 1160–1171, 1999.

[15] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 6(6):721–741, 1984.