

Classification using Intersection Kernel Support Vector Machines is Efficient

Subhransu Maji
Alexander C. Berg
Jitendra Malik

Presented by Nicholas Carlevaris-Bianco

Classification using Intersection Kernel Support Vector Machines is Efficient

- For Non-Linear **Histogram Intersection Kernels**
 - Can drastically reduce computational cost
 - Naively $O(nm)$
 - Provide exact solution in $O(n \log m)$
 - Provide approximate solution on $O(n)$ (*same as linear SVM*)
- Up to **2000x** faster on data set with large number of support vectors

n = dimension feature vectors

m = support vectors

Review of Support Vector Machines

- Binary Case

Input space

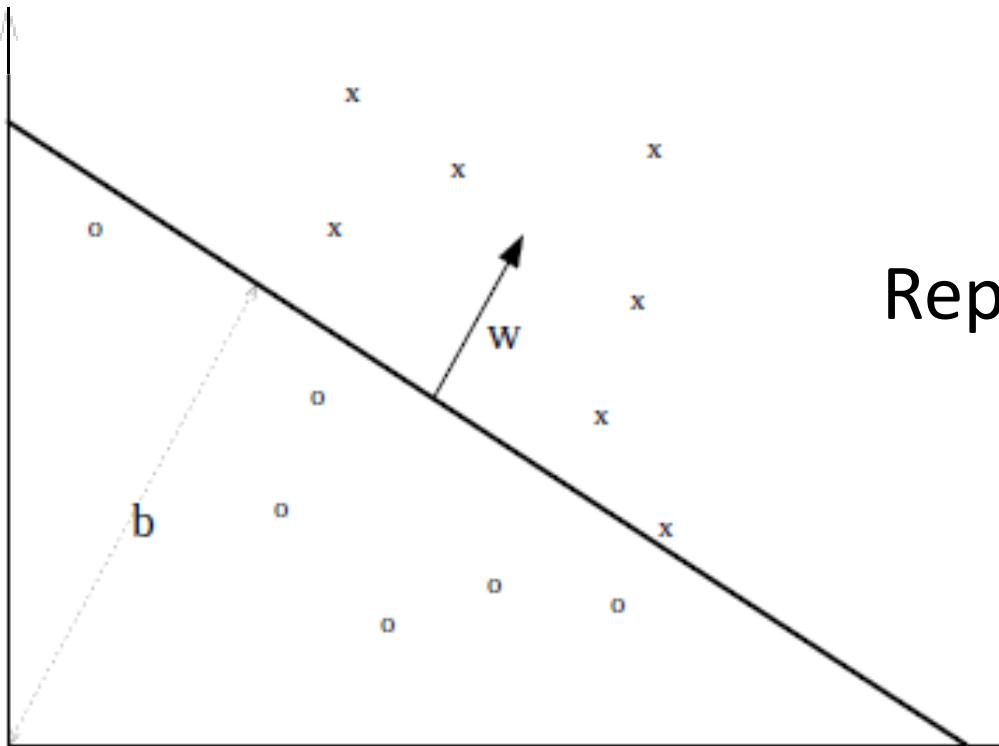
$$x \in X$$

Output space

$$y \in Y = \{-1, +1\}$$

Training Set

$$S = \{(x_1, y_1), \dots, (x_i, y_i), \dots\}$$

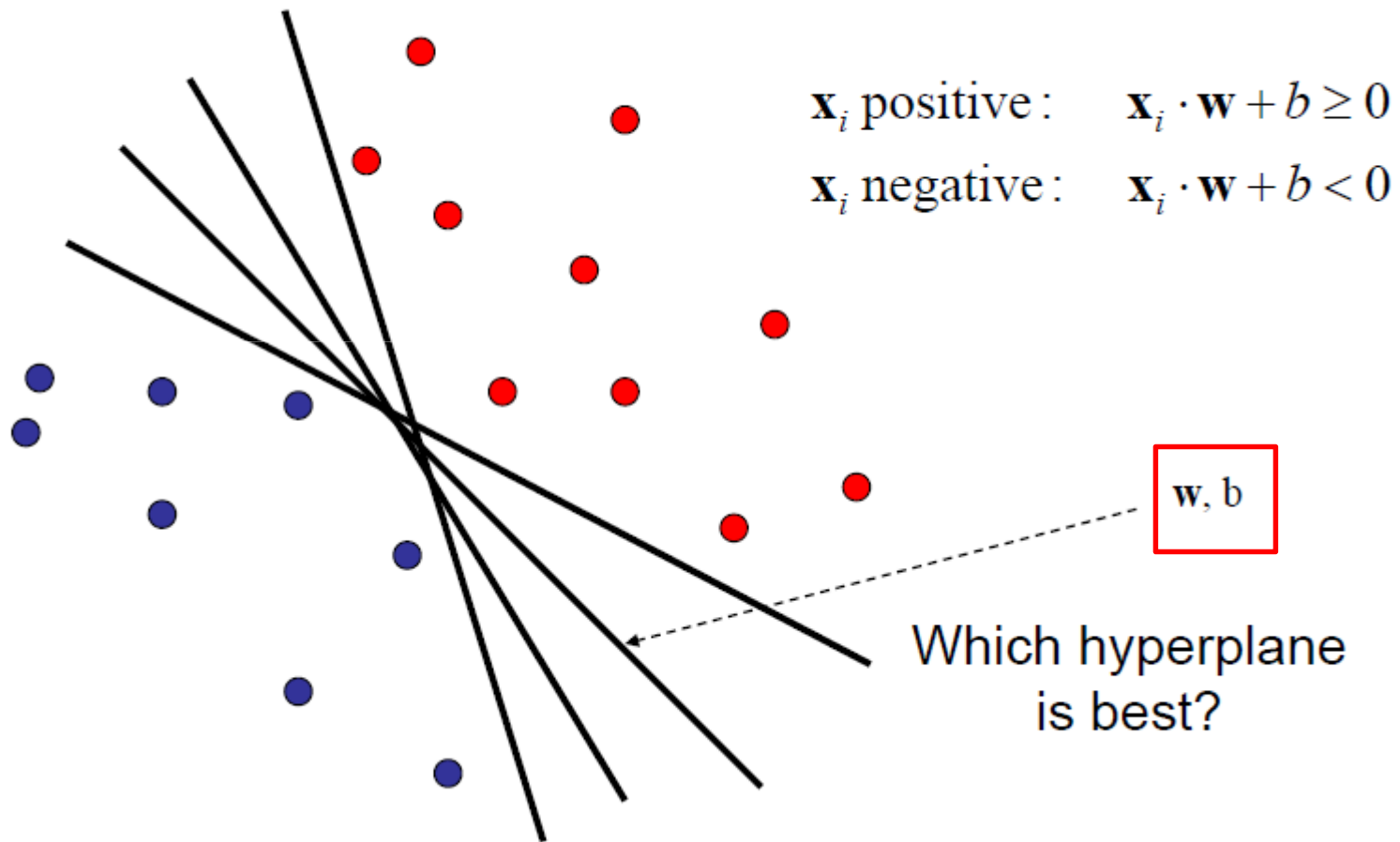


Representing a Hyperplane

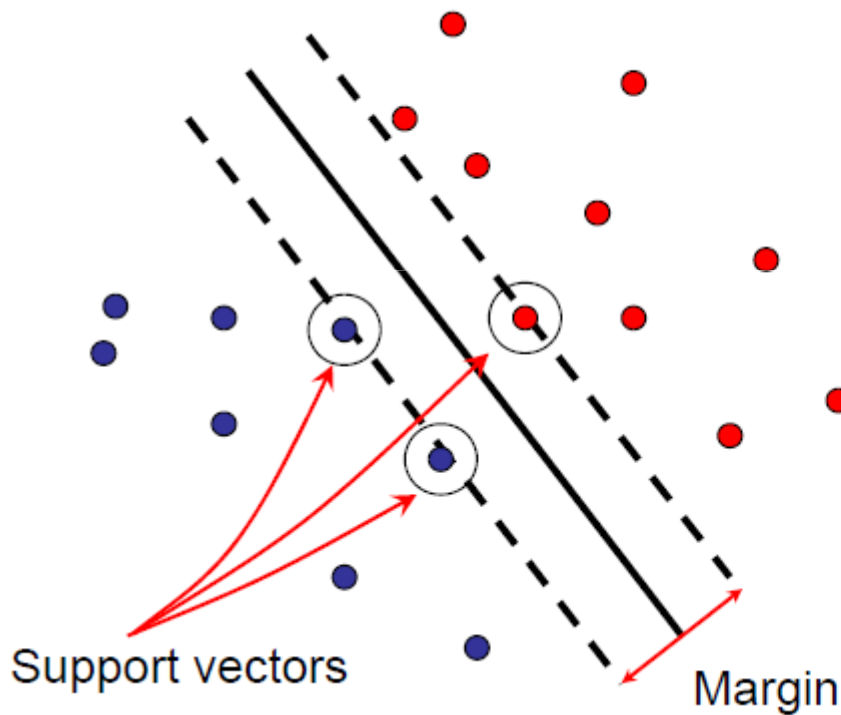
$$f(x) = \langle w, x \rangle + b$$

$$h(x) = \text{sign}(f(x))$$

Review of Support Vector Machines



Review of Support Vector Machines



$$\mathbf{x}_i \text{ positive } (y_i = 1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$$

$$\mathbf{x}_i \text{ negative } (y_i = -1): \quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$$

$$\text{For support, vectors, } \mathbf{x}_i \cdot \mathbf{w} + b = \pm 1$$

$$\text{Distance between point and hyperplane: } \frac{|\mathbf{x}_i \cdot \mathbf{w} + b|}{\|\mathbf{w}\|}$$

$$\text{Therefore, the margin is } 2 / \|\mathbf{w}\|$$

Review of Support Vector Machines

1. Maximize margin $2/\|\mathbf{w}\|$
2. Correctly classify all training data:

$$\mathbf{x}_i \text{ positive } (y_i = 1) : \quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$$

$$\mathbf{x}_i \text{ negative } (y_i = -1) : \quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$$

- *Quadratic optimization problem:* w = linear combination of training data

- Minimize $\frac{1}{2} \mathbf{w}^T \mathbf{w}$

$$w = \sum \alpha_i y_i X_i$$

Subject to $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1$

Minimized over *alpha* and *b*

Review of Support Vector Machines

- Now we have w in decision function

$$f(x) = \langle w, x \rangle + b$$

$$h(x) = \text{sign}(f(x))$$

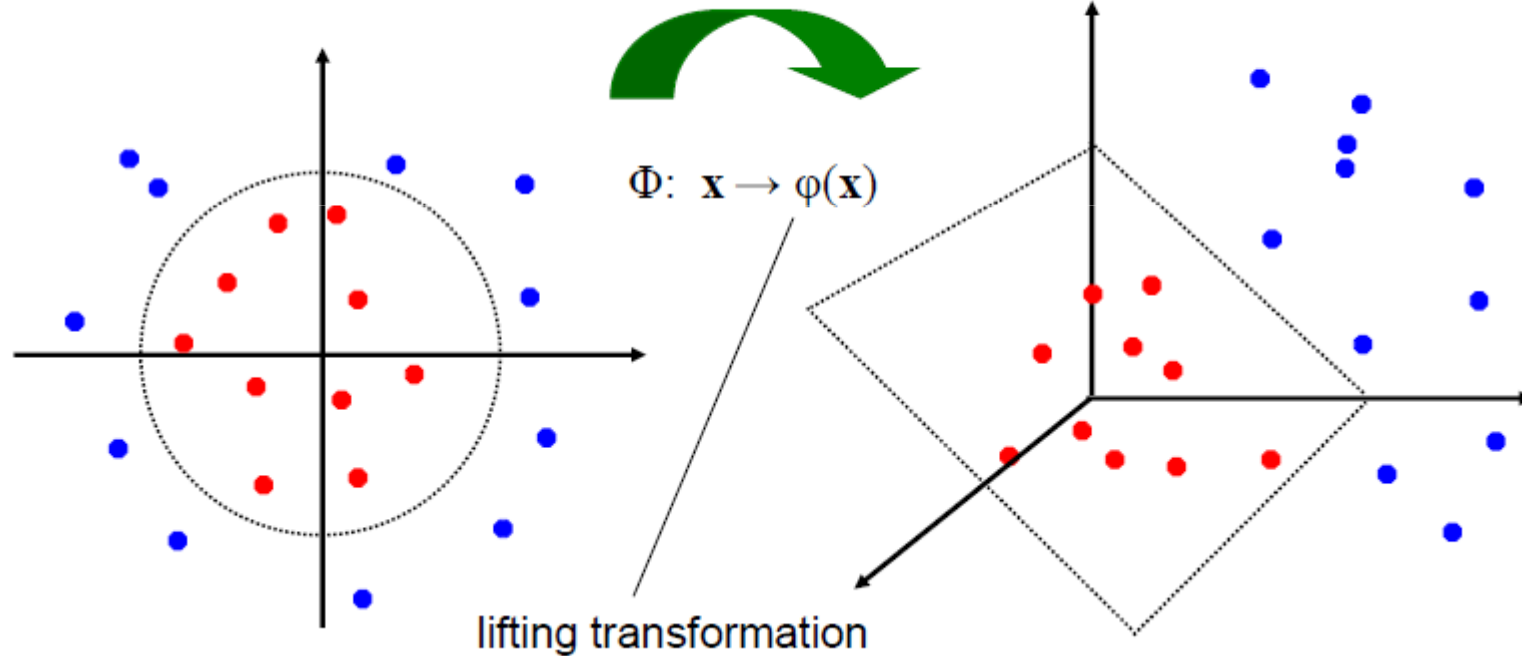
- Expand

$$w = \sum \alpha_i y_i x_i$$

$$f(x) = \langle w, x \rangle + b = \sum \alpha_i y_i \langle x_i, x \rangle + b$$

Review of Support Vector Machines

- Non-linear



- Problem: don't know $\varphi(\mathbf{x})$

Review of Support Vector Machines

- Mercer's Theorem

Inner product in higher dimensional space

$$K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)$$

- If K positive definite

$$\int K(x, z) f(x) f(z) dx dz \geq 0$$
$$\forall f \in L_2$$

- We don't need to know $\varphi(\mathbf{x})$ only that K is pos. def.

$$f(x) = \sum \alpha_i y_i \underbrace{\langle \phi(x_i), \phi(x) \rangle}_{K(\mathbf{x}_i, \mathbf{x}_j)} + b$$

Histogram Intersection Kernel

- Two histograms, each with n bins.

$$k_{HI}(h_a, h_b) = \sum_{i=1}^n \min(h_a(i), h_b(i))$$

- In "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories"
 - pyramid match kernel is a weighted sum of histogram intersections

Speeding Up Histogram Intersection Kernel SVM

- Classification function

n = dimension feature vectors

m = support vectors

$$\begin{aligned}h(\mathbf{x}) &= \sum_{l=1}^m \alpha_l y_l k(\mathbf{x}, \mathbf{x}_l) + b \\ &= \sum_{l=1}^m \alpha_l y_l \left(\sum_{i=1}^n \min(x(i), x_l(i)) \right) + b\end{aligned}$$

- Naïve implementation $O(nm)$

Speeding Up Histogram Intersection Kernel SVM

- Exact solution in $O(n \log m)$

$$\begin{aligned} h(\mathbf{x}) &= \sum_{l=1}^m \alpha_l y_l \left(\sum_{i=1}^n \min(x(i), x_l(i)) \right) + b \\ &= \sum_{i=1}^n \left(\sum_{l=1}^m \alpha_l y_l \min(x(i), x_l(i)) \right) + b \\ &= \sum_{i=1}^n h_i(x(i)) + b \end{aligned}$$

$$h_i(s) = \sum_{l=1}^m \alpha_l y_l \min(s, x_l(i))$$

Speeding Up Histogram Intersection Kernel SVM

- Exact solution in $O(n \log m)$

The i^{th} bin

Consider the functions $h_i(s)$ for a fixed value of i . Let $\bar{x}_l(i)$ denote the sorted values of $x_l(i)$ in increasing order with corresponding α 's and labels as $\bar{\alpha}_l$ and \bar{y}_l . If $s < \bar{x}_1(i)$ then $h_i(s) = 0$, otherwise let r be the largest integer such that $\bar{x}_r(i) \leq s$. Then we have,

$$\begin{aligned} h_i(s) &= \sum_{l=1}^m \bar{\alpha}_l \bar{y}_l \min(s, \bar{x}_l(i)) \\ &= \sum_{1 \leq l \leq r} \bar{\alpha}_l \bar{y}_l \bar{x}_l(i) + s \sum_{r < l \leq m} \bar{\alpha}_l \bar{y}_l \\ &= A_i(r) + s B_i(r) \end{aligned}$$

Piecewise linear

Speeding Up Histogram Intersection Kernel SVM

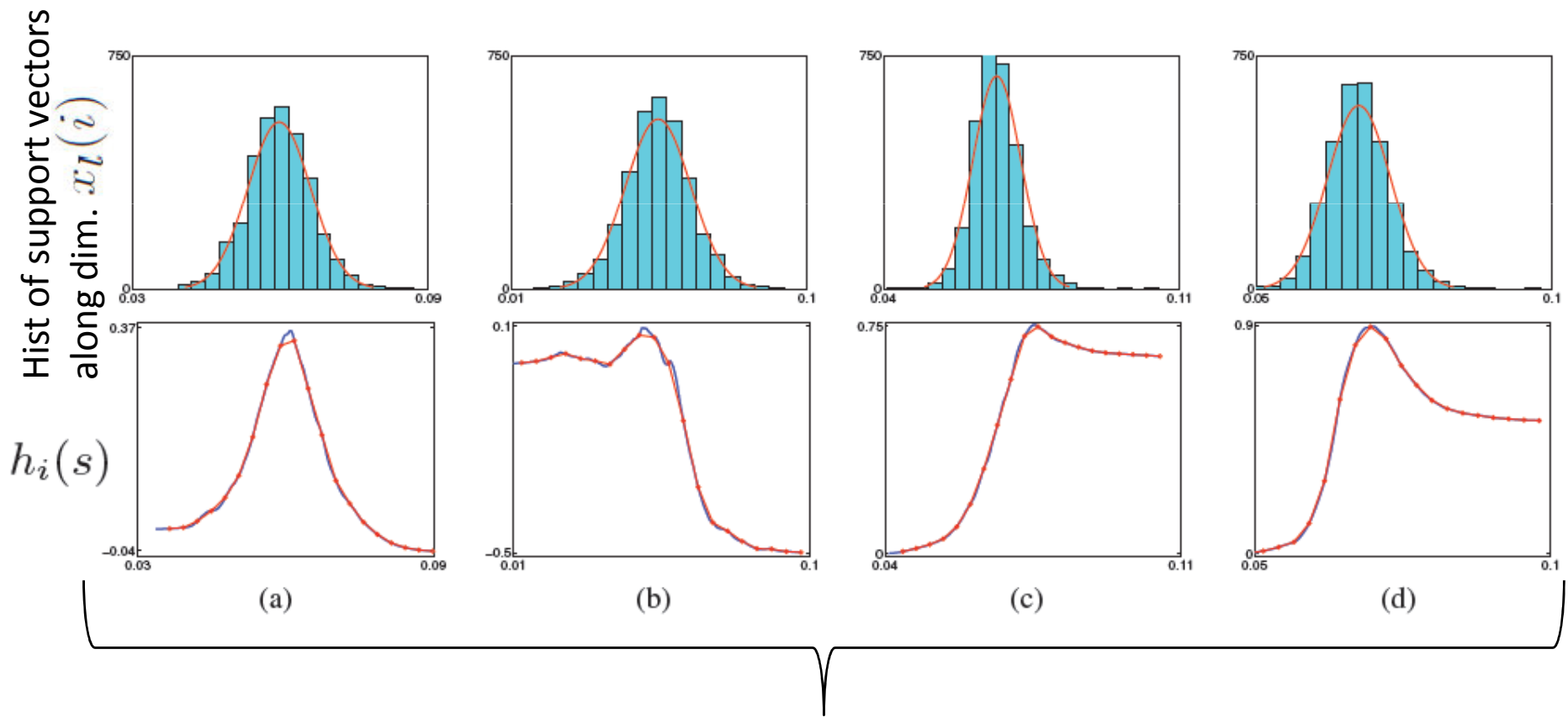
- Exact solution in $O(n \log m)$

$$\begin{aligned}h_i(s) &= \sum_{l=1}^m \bar{\alpha}_l \bar{y}_l \min(s, \bar{x}_l(i)) \\ &= \sum_{1 \leq l \leq r} \bar{\alpha}_l \bar{y}_l \bar{x}_l(i) + s \sum_{r < l \leq m} \bar{\alpha}_l \bar{y}_l \\ &= \boxed{A_i(r) + sB_i(r)}\end{aligned}$$

- A and B not dependent on data, can pre-compute.
- Computation $O(n \log m)$, memory $O(nm)$

Speeding Up Histogram Intersection Kernel SVM

- Approximate solution in $O(n)$



4 Sample Feature Space Dimensions

Speeding Up Histogram Intersection Kernel SVM

- Approximate solution in $O(n)$
- Assume piecewise **Linear** or **Constant**
- Sample $h_i(s)$ and pre-compute in lookup table.
- 30-50 samples enough to prevent loss of accuracy
- With large number of support vectors (5000) this provides a huge savings in memory

Speeding Up Histogram Intersection Kernel SVM

- Generalizes to any Kernel of the form

$$K_{GHI}^f(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^n \min(f(x(i)), f(z(i)))$$

Experimental Results

- Pedestrian detection
- Setup, using multi-level oriented Histogram of Gradients as feature descriptor

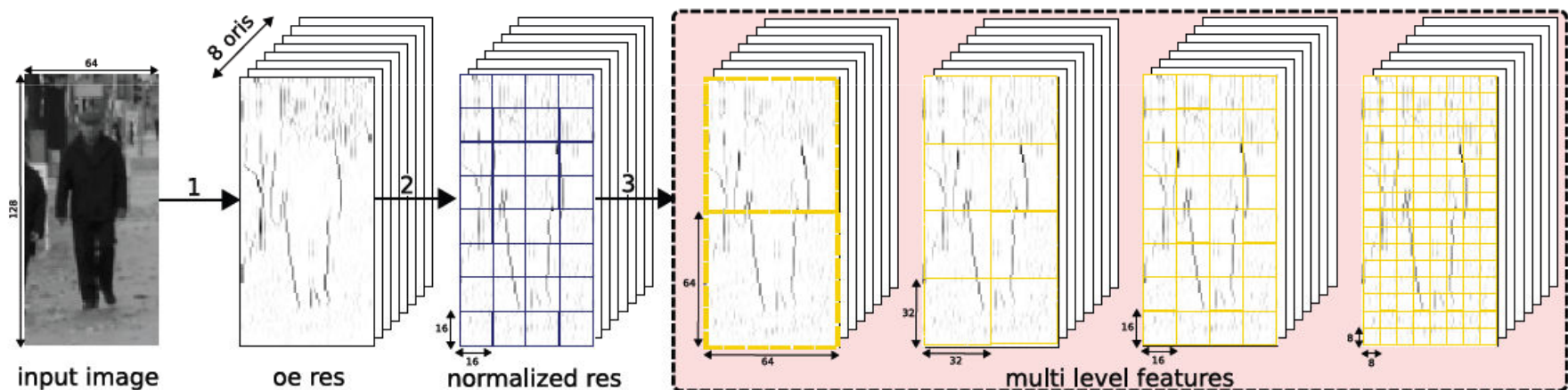
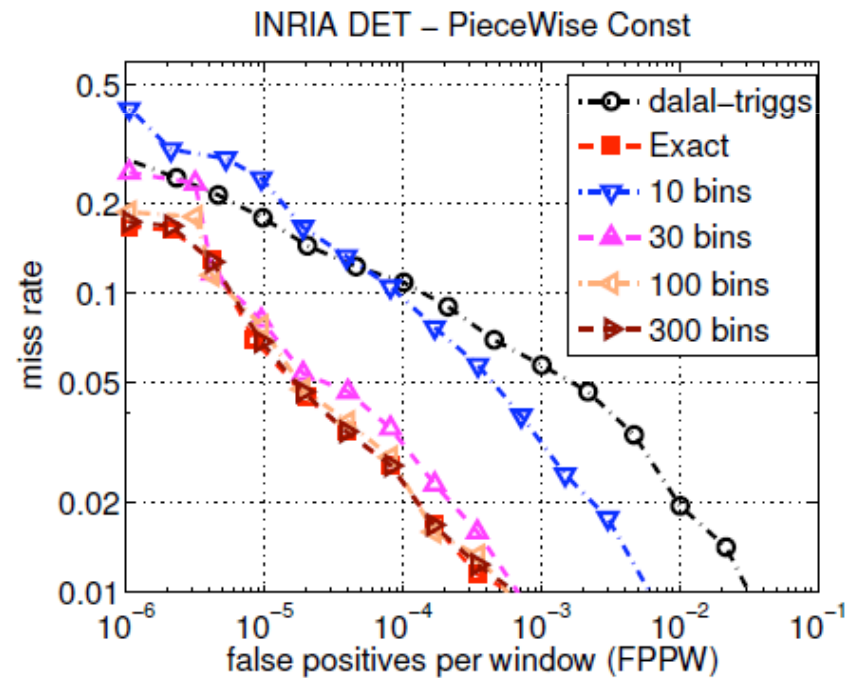
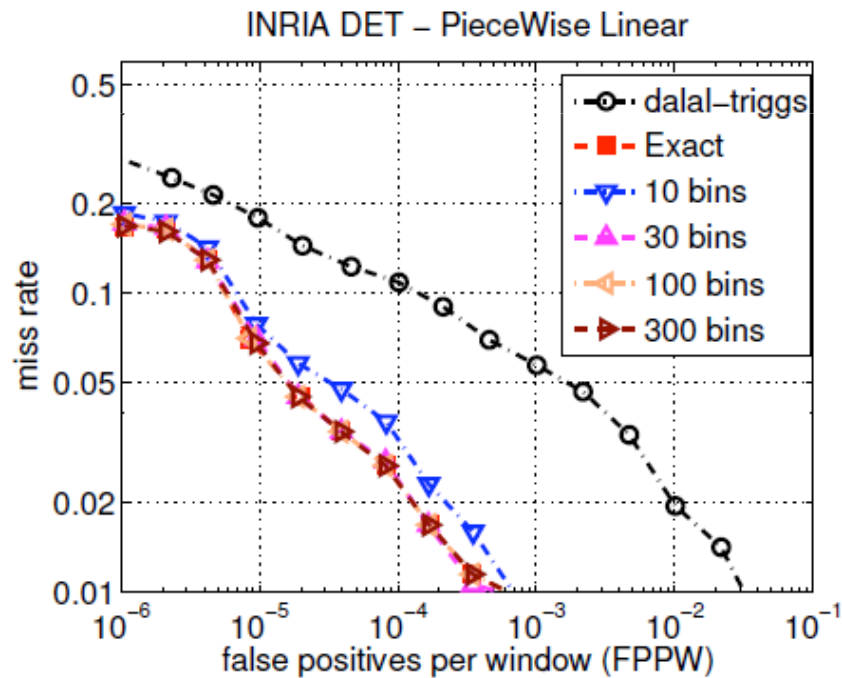


Figure 2. The three stage pipeline of the feature computation process. (1) The input grayscale image of size 64×128 is convolved with oriented filters ($\sigma = 1$) in 8 directions, to obtain oriented energy responses. (2) The responses are then L_1 normalized over all directions in each non overlapping 16×16 blocks independently to obtain normalized responses. (3) Multilevel features are then extracted by constructing histograms of oriented gradients by summing up the normalized response in each cell. The diagram depicts progressively smaller cell sizes from 64×64 to 8×8 .

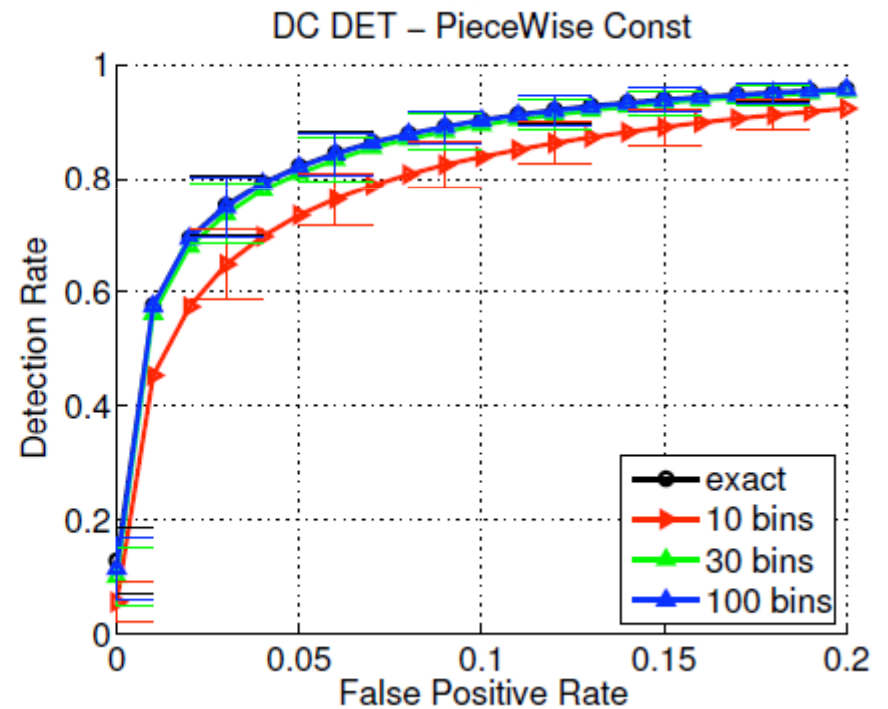
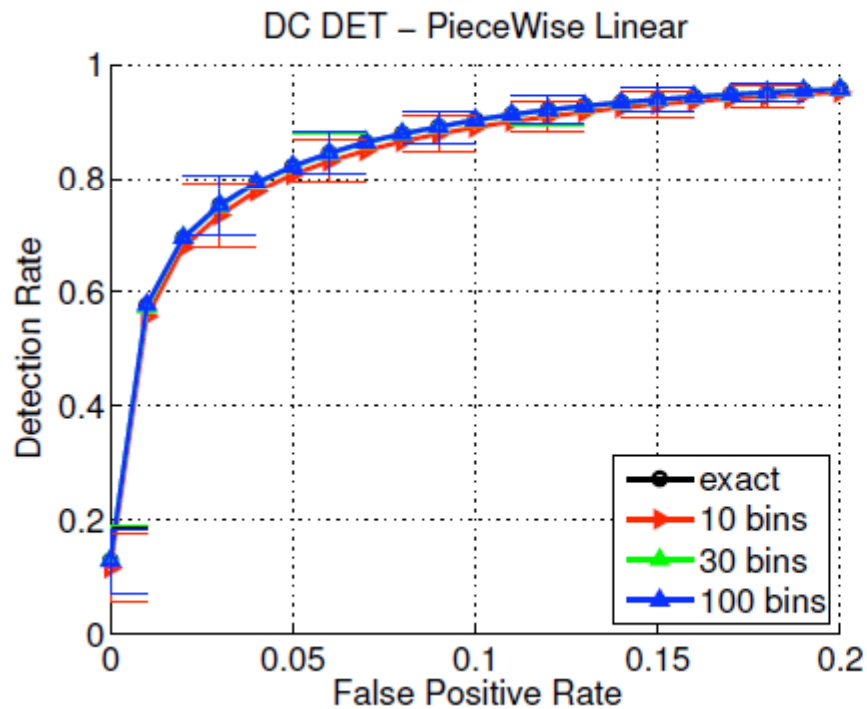
Experimental Results

- INRIA pedestrian dataset



Experimental Results

- INRIA pedestrian dataset



Experimental Results

- Speed over, INRIA pedestrian, Daimler Chrysler pedestrian and Caltech 101 object datasets

Dataset	Model parameters		SVM kernel type		fast IKSVMs		
	#SVs	#features	linear	intersection	binary search	piecewise-const	piecewise-lin
INRIA Ped	3363	1360	0.07±0.00	659.1±1.92	2.57±0.03	0.34±0.01	0.43±0.01
DC Ped	5474±395	656	0.03±0.00	459.1±31.3	1.43±0.02	0.18±0.01	0.22±0.00
Caltech 101	175±46	1360	0.07±0.01	24.77±1.22	1.63±0.12	0.33±0.03	0.46±0.03

$$\underbrace{\quad O(n) \quad O(nm) \quad O(n \log m) \quad O(n) \quad O(n) \quad}_{\text{Runtime in second}}$$