Beyond Hard Negative Mining: Efficient Detector Learning via Block-Circulant Decomposition

Alberto Bietti

December 18, 2013
Motivation

For training an object detector, usually:

- Train with all positives and some randomly sampled negatives
- Repeat:
  - Add high-scoring false positives from training images (*hard negatives*) to the training set
  - Retrain classifier
Motivation

For training an object detector, usually:

- Train with all positives and some randomly sampled negatives
- Repeat:
  - Add high-scoring false positives from training images (hard negatives) to the training set **Slow!**
  - Retrain classifier
Motivation

Problems:

- Finding hard negatives is *slow*: sliding window, multiple scales, multiple rounds
- A lot of useful data remains unused (and exhaustive search is prohibitively expensive)
Idea

- Windows inside an image are translated versions of each other.
- Constraints between these images reduces the degrees of freedom: exploit this to make learning easier.
Idea

- Windows inside an image are translated versions of each other
- Constraints between these images reduces the degrees of freedom: exploit this to make learning easier
- **How?** The Gram matrix of the data $G = (x_i^T x_j)_{ij}$ can be block-diagonalized
Why the Gram matrix

Many classifiers can be learned by solving

$$\min_w \|w\|^2 + C \sum_i^n L(w^\top x_i, y_i)$$

Dual formulation:

$$\min_\alpha \frac{1}{2} \alpha^\top G \alpha + \sum_i^n D(\alpha_i, y_i)$$
Why the Gram matrix

Many classifiers can be learned by solving

\[ \min_w \|w\|^2 + C \sum_i^n L(w^\top x_i, y_i) \]

Dual formulation:

\[ \min_\alpha \frac{1}{2} \alpha^\top G \alpha + \sum_i^n D(\alpha_i, y_i) \]

Only depends on the data through \( G \).
Figure 1: (a) An augmented training set of base samples, and horizontal translations. Such data arises when training a classifier with cyclic shifts. In the following sections, we will consider translations of an augmented training set $X$. For the training set in Eq. (5), we will see how $X$ can be expressed in the dual, in terms of a Gram matrix $G$. As mentioned in Section 2.1, many learning algorithms prove several subwindows from images.

Recall that each element of $G$ encodes the interaction between some groups of samples. This is proved in Section 3. If the interactions between some groups of samples are set to 0, we can invoke the Convolution Theorem to compute these correlations faster in the Fourier domain. Another advantage is that, since Eq. (6) to (8) compute these correlations faster in the Fourier domain.

**Illustration**

<table>
<thead>
<tr>
<th>Images</th>
<th>Translations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i = 1 \ldots n$</td>
<td>$\alpha = 1 \ldots s$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$G$</th>
</tr>
</thead>
</table>

Block-diagonalization consists of finding a matrix $U$ such that $\bar{G} = UGU^{-1}$ takes block-diagonal form: the original sample is recovered for each column to another sample with index $i$, $j$. The structure of $G$ is affected by translated samples. Since the blocks $G(i, j)$ to a pair of base samples (see Fig. (a)), $\bar{G}$ can be made more evident as $G$ to $\bar{G}$. This reveals that storing the full auxiliary vectors is not necessary, as only the auxiliary vectors with $\alpha = \beta$ required.

$$\bar{G} = UGU^{-1}$$
Augmented dataset $\mathcal{X} = \{ P^{u-1} x_i | i = 1, \ldots, n; u = 1, \ldots, s \}$

We have

$$G_{(u,v),(i,j)} = x_i^\top P^{v-u} x_j = g_{v-u}(i,j)$$

This gives correlations

$$g(i,j) = x_i \ast x_j = \mathcal{F}^{-1}(\mathcal{F}^*(x_i) \cdot \mathcal{F}(x_j))$$
New formulation

If \( G = U^{-1} \text{diag}(G_1, \ldots, G_s)U \),

\[
\min_{\alpha} \frac{1}{2} \alpha^\top G \alpha + \sum_{i}^{n} D(\alpha_i, y_i)
\]

is equivalent to the \( s \) independent problems:

\[
\min_{\alpha_f} \frac{1}{2} \alpha_f^\top G_f \alpha_f + \sum_{i}^{n} D(\alpha_{fi}, y_{fi}), \quad f = 1, \ldots, s
\]
Algorithm

Inputs:
- \( X \) (\( m \) features on a \( s_1 \times s_2 \) grid for \( n \) samples, total size \( s_1 \times s_2 \times m \times n \))
- \( Y \) (labels, size \( s_1 \times s_2 \times n \))
- regression (a linear regression function)

Output:
- \( W \) (weights, size \( s_1 \times s_2 \times m \))

\[
X = \text{fft2}(X) / \sqrt{s_1 \times s_2}; \quad Y = \text{fft2}(Y) / \sqrt{s_1 \times s_2};
\]
\[
Y(1,1,:) = 0;
\]
\[
X = \text{permute}(X, [4, 3, 1, 2]); \quad Y = \text{permute}(Y, [3, 1, 2]);
\]
\[\text{for } f_1 = 1:s_1 \text{ do}\]
\[\quad \text{for } f_2 = 1:s_2 \text{ do}\]
\[\quad\quad W(f_1,f_2,:) = \text{regression}( \ldots \text{, } X(:, :, f_1, f_2), Y(:, f_1, f_2) ) ;\]
\[\quad\end{eqnarray}\]
\[\end{eqnarray}\]
\[\text{end}\]
\[\end{eqnarray}\]
\[\text{end}\]
\[
W = \text{real}(\text{ifft2}(W)) * \sqrt{s_1 \times s_2};
\]
Experiments

Pedestrian detection, INRIA person dataset
Experiments
Results

(a) Randomly sampled (0.777)  
(b) 1 mining round (0.792)  
(c) 2 mining rounds (0.806)  
(d) Circulant decomposition (0.867)
Conclusions

Benefits of the method:

- Uses all the available image data
- Only one training phase, no slow hard-negative mining phase
- Easily parallelizable
Conclusions

Benefits of the method:

- Uses all the available image data
- Only one training phase, no slow hard-negative mining phase
- Easily parallelizable

Issues:

- Decomposition only works with some types of classifiers (SVR, Ridge regression), and filter-like features
- The training set can become quite large (small window at small scales)
References

J. F. Henriques, J. Carreira, R. Caseiro, J. Batista
*Beyond Hard Negative Mining: Efficient Detector Learning via Block-Circulant Decomposition*