Regularized Max Pooling for Image Categorization

Minh Hoai
http://www.robots.ox.ac.uk/~minhhoai/

Visual Geometry Group, Department of
Engineering Science, Oxford University

We propose Regularized Max Pooling (RMP) for image classification. RMP classifies an image (or image region) by extracting feature vectors at multiple subwindows at multiple locations and scales. Unlike Spatial Pyramid Matching where the subwindows are defined purely based on geometric correspondence, RMP accounts for the deformation of discriminative parts. The amount of deformation and the discriminative ability for multiple parts are jointly learned during training.

An RMP model is a collection filters. Each filter is anchored to a specific image subwindow and associated with a set of deformation coefficients. The anchoring subwindows are predetermined at various locations and scales, while the filters and deformation coefficients are learnable parameters of the model. Fig. 1 shows a possible way to define subwindows. To classify a test image, RMP extracts feature vectors for all anchoring subwindows. The classification score of an image is the weighted sum of all filter responses. Each filter yields a set of filter responses, one for each level of deformation. The deformation coefficients are the weights for these filter responses.

Given a set of images $\{\mathbf{I}_i\}_{i=1}^n$ and labels $\{y_i | y_i \in \{1,-1\}\}_{i=1}^n$, consider a particular set of geometrically defined subwindows which can encode semantic content of an image at different locations and scales (e.g., Fig 1). Let $\{\mathbf{I}^j\}_{j=1}^m$ denote the set of subwindows for image $\mathbf{I}$. Let ϕ be the feature function of which the input is an image region and the output is a column vector. Let $\mathbf{D}^j$ be the feature matrix computed at location j for all images and $\mathbf{K}^j$ the corresponding kernel, i.e., $\mathbf{D}^j = [\phi(\mathbf{I}_1^j) \cdots \phi(\mathbf{I}_n^j)]$ and $\mathbf{K}^j = (\mathbf{D}^j)^T \mathbf{D}^j$. The joint kernel for all subwindows is the sum of all kernels: $\mathbf{K} = \sum_{j=1}^m \mathbf{K}^j$; this corresponds to concatenating all feature vectors computed at all subwindows. Given the kernel $\mathbf{K}$, we train an Least-Squares SVM and obtain a coefficient vector and bias term α, b. The filter for subwindow j can be computed as $\mathbf{w}^j = \mathbf{D}^j \alpha$.

For a particular subwindow j and an image $\mathbf{I}$, the regularized maximum score is defined:

$$f^j(\gamma) = \max_{k \in \{1,\cdots,m\}} \left\{ (\mathbf{w}^j)^T \phi(\mathbf{I}^k) - \gamma . dist(\mathbf{I}^k, \mathbf{I}^j) \right\}. \quad (1)$$

Here γ is a non-negative regularization parameter and $dist(\cdot,\cdot)$ is the square geometric distance between two regions. The square geometric distance from a region R' to a reference region R is defined as:

$$dist(R', R) = \left(\frac{x'-x}{w}\right)^2 + \left(\frac{y'-y}{h}\right)^2 + \log_2^2\left(\frac{w'}{w}\right) + \log_2^2\left(\frac{h'}{h}\right), \quad (2)$$

where (x, y, w, h) and (x', y', w', h') are the center locations, the widths, and the heights of regions R and R' respectively. This distance function is asymmetric. It is invariant to the scale of the coordinate system. The last two terms of Eq. (2) measure the scale distance between R' and R. We use $\log_2(\cdot)$ to ensure that the scale distance from R' to R is the same for the following two cases: (i) R' is k times bigger than R; (ii) R' is k times smaller than R.

The value of $f^j(\gamma)$ is the regularized maximum response; it seeks a location with high filter response and low deformation cost w.r.t. to the anchor region $\mathbf{I}^j$. If γ is 0, $f^j(\gamma)$ is the maximum filter response. If γ is big, $\gamma . dist(\mathbf{I}^k, \mathbf{I}^j)$ will be big except for $k = j$ where $dist(\mathbf{I}^j, \mathbf{I}^j) = 0$. Thus, for a big γ, $f^j(\gamma) = (\mathbf{w}^j)^T \phi(\mathbf{I}^j)$, which is the filter response of the anchor region.

The right setting for γ depends on the level of deformation of region j of the semantic class in consideration. Since the deformation level of a region is unknown, we start with an over-complete set of γ's and learn the tradeoff between deformation and discrimination. For each region j of an image $\mathbf{I}$, we construct a feature vector by varying the value of $\gamma \in \{\gamma_1, \cdots, \gamma_k\}$ and compute the regularized maximum response. Let $\mathbf{f}^j$ be the vector of obtained values, i.e., $\mathbf{f}^j = [f^j(\gamma_1), \cdots, f^j(\gamma_k)]^T$. For each image, we obtain a feature matrix by accumulating the filter responses for all regions $\mathbf{F} = [\mathbf{f}^1 \cdots \mathbf{f}^m]$. Let $\mathbf{F}_i$ be the feature matrix for image $\mathbf{I}_i$. We jointly learn the deformation and discriminative ability of all regions by solving the following optimization problem:

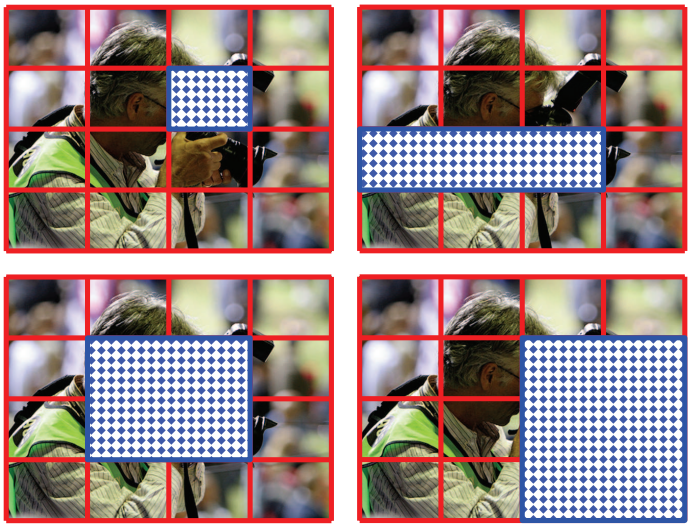

Figure 1: **From grid division to subwindows**. An image is divided into 4×4 blocks. We consider rectangular subwindows that can be formed by a contiguous chunk of blocks. There are 100 such subwindows, and this figure shows four examples.

$$\underset{\mathbf{S}, \overline{b}}{\text{minimize}} \sum_{i=1}^n (trace(\mathbf{S}^T \mathbf{F}_i) + \overline{b} - y_i)^2 \quad (3)$$

$$\text{s.t. } s_{lj} \geq 0 \; \forall l = 1, \cdots, k, \; \forall j = 1, \cdots, m. \quad (4)$$

The above optimizes over a weight matrix $\mathbf{S} \in \Re^{k \times m}$ and a bias term $\overline{b}$. Each column of $\mathbf{S}$ is a weight vector for a particular region; it learns weights for the regularized maximum responses for different values of γ's. The weights should be non-negative to emphasize the relative importance of higher filter responses. The objective of the above formulation minimizes the sum of L_2 losses.

We start with an over-complete set of γ's and let the algorithm determines the right level of allowable deformation. In our experiments, we use $\gamma_1 = 0$, $\gamma_k = \infty$, $\gamma_l = 2^l / 10^4$ for $l = 2, \cdots, k-1$, with $k = 15$. The feasible set of $\mathbf{S}$ is suitable for different levels of deformation, including the following two extreme cases:

1. Well-aligned semantic concept. For an image categorization task where the semantic concepts are well aligned, rigid geometric alignment is the right model. In this case, the weight matrix $\mathbf{S}$ could be all zeros except for the last row of all ones (the last row corresponds to $\gamma = \infty$).
2. Highly deformed semantic concept. For categorization tasks where the semantic concepts have high level of deformation, geometric correspondence should be ignored. In this case, the weight matrix $\mathbf{S}$ could be all zeros except for the first row of all ones (the first row corresponds to $\gamma = 0$).

This formulation corresponds to a linear program, which can be optimized efficiently using a linear programming solver such as Cplex.

We demonstrate the benefits of RMP in recognizing human actions in still images. RMP outperforms Deformable Part Models and Spatial Pyramid Matching, especially for action classes with high level of deformation. Furthermore, the simplicity and flexibility of RMP allow it to be used with any type of features, including Convolutional Neural Network (CNN) features. Together with CNN features, RMP establishes the new state-of-the-art performance for human action recognition in still images, evaluated on the challenging dataset of PASCAL VOC 2012.