Computer Science and Artificial Intelligence Laboratory
Technical Report

MIT-CSAIL-TR-2012-032 October 1, 2012

A Gaussian Approximation of Feature Space for Fast Image Similarity

Michael Gharbi, Tomasz Malisiewicz, Sylvain Paris, and FrØdo Durand

massachusetts institute of technology, cambridge, ma 02139 usa — www.csail.mit.edu

A Gaussian Approximation of Feature Space for Fast Image Similarity

Michael Gharbi
MIT CSAIL
gharbi@mit.edu

Tomasz Malisiewicz
MIT CSAIL
tomasz@csail.mit.edu

Sylvain Paris
Adobe
sparis@adobe.com

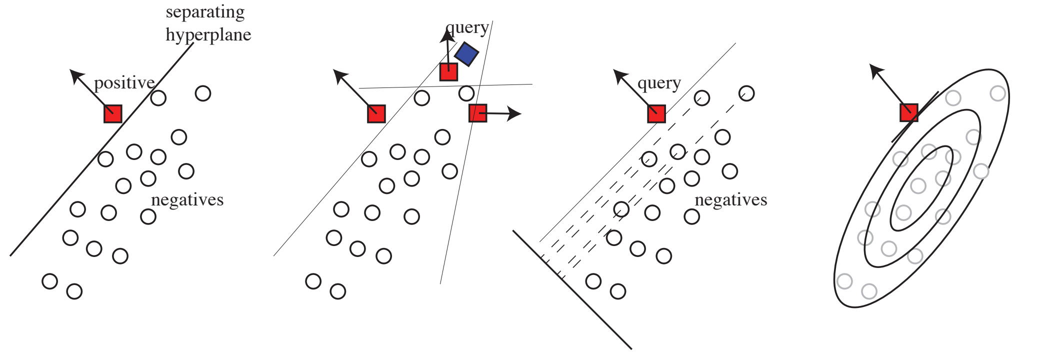
Frédo Durand
MIT CSAIL
fredo@mit.edu

Abstract

We introduce a fast technique for the robust computation of image similarity. It builds on a re-interpretation of the recent exemplar-based SVM approach, where a linear SVM is trained at a query point and distance is computed as the dot product with the normal to the separating hyperplane. Although exemplar-based SVM is slow because it requires a new training for each exemplar, the latter approach has shown robustness for image retrieval and object classification, yielding state-of-the-art performance on the PASCAL VOC 2007 detection task despite its simplicity. We re-interpret it by viewing the SVM between a single point and the set of negative examples as the computation of the tangent to the manifold of images at the query. We show that, in a high-dimensional space such as that of image features, all points tend to lie at the periphery and that they are usually separable from the rest of the set. We then use a simple Gaussian approximation to the set of all images in feature space, and fit it by computing the covariance matrix on a large training set. Given the covariance matrix, the computation of the tangent or normal at a point is straightforward and is a simple multiplication by the inverse covariance. This allows us to dramatically speed up image retrieval tasks, going from more than ten minutes to a single second. We further show that our approach is equivalent to feature-space whitening and has links to image saliency.

1 Introduction

The core task of matching entire images or windows containing an object is at the heart of a variety of algorithms in computer vision and image retrieval, including object recognition, scene classification, internet image search, or data-driven in-painting. Retrieval tasks seek to find the image that is most similar to a query, and the challenge is that simple metrics such as the sum of square differences of pixel do not correspond to human's notions of similarity, especially when people expect "semantic" matches. For object-class detection, nearest-neighbor approaches search in a training set for the most similar exemplar to an input image window. In addition to classification, they have the benefit that the association between the query and a training datapoint enables the transfer of information beyond class. One drawback of nearest-neighbor approaches is that computer vision tasks usually require training datasets that include millions of images or windows in order to offer a comprehensive coverage of negative examples. In addition, the definition of appropriate metrics is challenging. These two problems have traditionally made it hard for nearest-neighbors to compete with state-of-the-art object detection techniques.



(a) SVM between one positive and many negatives
(b) Collection of exemplar-based SVM for classification
(c) Exemplar-based SVM for search
(d) Our approach approximates the training set by a Gaussian and computes closed-form normals

Figure 1: (a) Malisiewicz et al.'s exemplar-based SVM trains a classifier for each query image and uses the normal to the hyperplane to rank other images. (b) Object classification for a query (blue diamond) is performed using nearest neighbor where the similarity to a positive exemplar is the dot product with its SVM normals. (c) For retrieval, images are ranked according to their dot product with an SVM trained at the query. (d) Our approach fits a Gaussian to a training database and computes the normal to the Gaussian in closed form by multiplying the query by the inverse covariance.

Recently, however, Malisiewicz et al. [14] introduced a technique that can be seen as an extension to nearest-neighbor classification, and in particular to the use of learned exemplar distance functions, e.g. [10, 8, 13], and that can handle large set of negatives. In order to define the distance to an exemplar, they uses a discriminative approach and train a linear SVM between the single positive exemplar and a large set of negative examples (Fig. 1(a) and (b)). The dot product with the SVM hyperplane's normal then provides a measure of similarity to the exemplar. The set of negatives is represented parametrically (by the individual SVMs), enabling this method to scale and yet yield explicit correspondences. Despite its simplicity, this method was shown to provide state of the art results on the PASCAL VOC detection task.

They then applied this approach to image search [16], where they train a linear SVM that discriminates between the query and a training datasets, and then use the dot product with the normal to the SVM hyperplane as a measure of similarity: images that have higher dot product are more similar (Fig. 1(c)). They argue that the SVM finds the direction in feature space that is most "unique" to the image and is therefore a good measure of similarity. They showed that the approach is robust and even enables cross-domain matches, e.g. between paintings and photographs. Unfortunately, the need to train an SVM against millions of negatives for each query and their technique takes more than 10 minutes each time.

In this paper, we revisit the exemplar-based SVM approach [14, 16] for image similarity and retrieval tasks. We dramatically accelerate it and shed new lights on the underlying mechanisms. Our work starts with the following interpretation of their technique. The linear SVM between a single point and a full set of negatives (Fig. 1a essentially computes the tangent and normal to the manifold of negatives at the positive. This might appear odd because Fig. 1a would suggest that this is only valid for points at the periphery of the manifold. In fact, the whole approach by Malisiewicz et al. [14] should work well only for points at the periphery of the space because they need each exemplar to be separable enough from the negatives. This apparent paradox is due to our limited intuition about high-dimensional spaces such as that induced by the Histogram of Oriented Gradients (HOG) they use as features, which comprises thousands of dimensions. In such a space, all points tend to lie at the periphery, which is one aspect of the curse of dimensionality [1, 2], or one of its blessings [5]. We discuss properties of high-dimensional spaces and move on to making the computation of normals to the manifold of images faster. For this, we fit a Gaussian by computing the covariance of million of training images. Given the covariance matrix, computing the normal to the Gaussian is straightforward. We show that this approach performs similarly to Malisiewicz et al.'s exemplar-based SVM on image retrieval and object detection tasks. More importantly, it is orders of magnitude faster, taking a similarity query from ten minutes to one second. Finally, we show that our approach is in fact equivalent to Linear Discriminant Analysis and to the whitening of the feature space, e.g. [6], which shed new light on the notion of image uniqueness and relates it to models of saliency and pop out phenomena [15]. We focus on the single-image query task, where we are given an image as input and seek to retrieve the most similar image or sub-window from a large set. The extension to object category based on an ensemble of positive examples requires additional machinery such as calibration but it follows from [14].

In concurrent work, Hariharan et al.[9] propose whitening HOG features for recognition tasks, which is similar to our approach. They focus more on a probabilistic perspective and Linear Discriminant Analysis, while we start from a re-interpretation of exemplar-based SVMs. They also propose an additional step for the object detection task during which they cluster the positive exemplars before applying them to the testing dataset. This provides an additional speed-up to the computation (because it effectively reduces the number of detections to perform) and consolidates the scores (for exemplars that output mostly false positives are tuned down in the process).

1.1 Background on examplar-based SVM

Our work is inspired by and builds upon the eSVM method of Malisiewicz et al. [12, 14, 16]. We summarize the main steps of the method hereafter and refer to their articles for the details.

The Exemplar-SVM method [12, 14, 16] learns a query-specific weight vector $\mathbf{w} \in \mathbb{R}^d$ for a query image $\mathbf{q} \in \mathbb{R}^d$, where images are represented as high-dimensional Histogram of Oriented Gradients (HOG) feature vectors [4]. The learned weights give rise to a query-specific similarity function $f_\mathrm{e}(\mathbf{x}) = \mathbf{w}_\mathrm{e}^\mathsf{T}\mathbf{x}$ which scores input vectors $\mathbf{x}$ based on a learned notion of similarity to the exemplar $\mathbf{q}$. Learning $\mathbf{w}_\mathrm{e}$ amounts to solving a standard L_2-regularized linear SVM objective using the query $\mathbf{q}$ as a single positive and the set $\mathcal{N}$ comprising millions of image vectors as negatives. Because the set of negatives includes all subwindows extracted from a large collection of images, it cannot be stored in memory. The method alternates between learning $\mathbf{w}_\mathrm{e}$ given an active set of negatives and mining new negatives in a sliding-window fashion (this technique is known as hard-negative mining [7]). The eSVM solution $\mathbf{w}_\mathrm{e}$ is the result of the following convex optimization problem:

$$\mathbf{w}_\mathrm{e} = \operatorname*{argmin}_{\mathbf{w}} \frac{\lambda_\mathrm{e}}{2}\|\mathbf{w}\|^2 + L(\mathbf{w}^\mathsf{T}\mathbf{q}) + \sum_{i \in \mathcal{N}} L(-\mathbf{w}^\mathsf{T}\mathbf{x}_i) \tag{1}$$

2 High dimensions and the HOG feature space

We first review general properties of high-dimensional spaces. Then, we study the HOG feature vectors and show that they exhibit the same properties.

The curse of dimensionality, distance, and separability The intuition we gain from low dimensional spaces about point distribution, distance and separability unfortunately do not extend to high dimensional spaces. This is one of the aspects of the so-called *curse of dimensionality* [1, §5.16] [2, §1.4] [5]. Consider the example of a large set of points drawn from a Gaussian distribution. In 2D, we picture a large concentration around the center, and fewer and fewer points as we increase the distance to the center. However, this arrangement changes when the dimensionality increases. The volume between the spheres of radii r and $r + \Delta r$ grows as r^{d-1}. Because of this, in higher dimensions, the center is comparatively smaller than the outer rings. Although the point density decreases with the distance, the volume increase dominates and most of the points are located at the periphery of the cloud as shown in Figure 2a. In such configuration, all points are on the convex hull of the point cloud or close it and it is possible to separate from the rest of the samples with a hyperplane.

HOG feature space To confirm that the HOG feature space exhibits the same properties as a generic high-dimensional space, we measured the density of samples as function of their norm and we varied the dimensionality by varying the number of spatial bins. Figure 2b shows that the HOG vectors have also tend to lie at the periphery of the space. As the dimensionality increases, the points cluster farther from the center. We confirmed this structure by measuring how many points can be isolated with an hyperplane. As the dimension increases, almost all the points fall in this category (Fig. 2c). This configuration where the feature vectors lie on the border of the space explains and is critical to the success of the eSVM technique. In the next section, we build upon this theoretical underpinning and design an efficient algorithm that exploits this structure to drastically reduce the computational cost of estimating the separating hyperplane.

3 Parametric approach

We have established that, in the high-dimensional space of HOG features, most data points lie at the outer hypersurface of the set of images. This allows us to interpret the exemplar-based SVM approach by Malisiewicz et al. [14] and Shrivastava et al. [16] roughly as the computation of the tangent and normal to this hypersurface at the query point. Indeed, for a convex set, the tangent is the plane that best separates a point from the rest of the set. This computation is expensive because SVM needs to find good negatives in the vast training set, essentially looking for neighbor to perform some kind of finite-difference normal approximation.

Our approach is dramatically faster and more robust because we fit a parametric model, a Gaussian, to the training set to approximate the manifold of images. Computing the normal to the Gaussian at a point is then straightforward. In this paper, we focus on the single-image query case and train our covariance on a generic set of images.

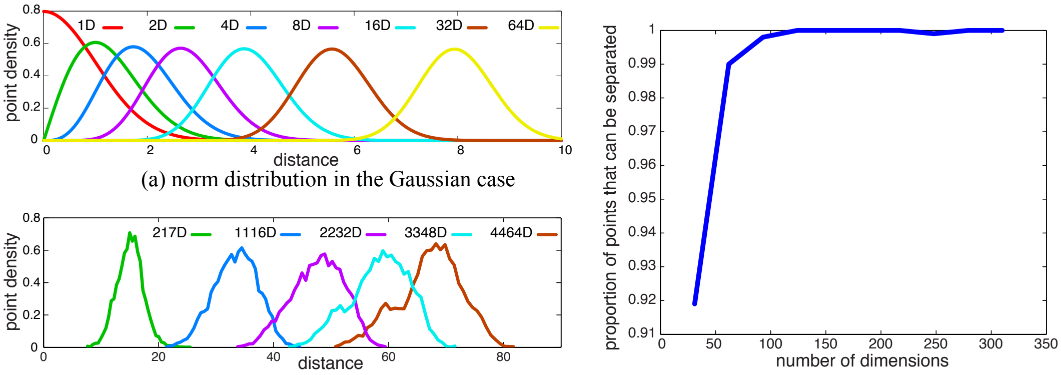
To fit the Gaussian, we compute the covariance matrix of millions of training images or windows, the negative set in the parlance of Malisiewicz et al. [14]. This set is independent of the query image (or object category) and needs to be computed only once. That is, whereas their representation of negatives (the SVM) is parametric, it is query-specific, whereas our parametric representation is general and shared by all queries.

We use the standard procedure to estimate the covariance matrix $\Sigma = \frac{1}{n}\sum_i(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\mathsf{T}$ where n is the number of feature vectors $\mathbf{x}_i$, and $\boldsymbol{\mu}$ their mean. Following Malisiewicz's later work [12] and Shrivastava et al. [16], we do not seek to exclude positives from the negative training set, since even for object detection, it has been shown that the relatively tiny number of positives has limited impact on the negative representation [12].

At run time, given a query image $\mathbf{q}$, we compute the normal to the Gaussian at $\mathbf{q}$ using the covariance matrix:

$$\mathbf{w} = \Sigma^{-1}(\mathbf{q} - \boldsymbol{\mu}). \quad (2)$$

This simple matrix-vector multiplication is relatively cheap (the matrix is, however, dense and has thousands of dimensions). This normal corresponds to the weight vector $\mathbf{w}$ used by Malisiewicz et al. [14].



(a) norm distribution in the Gaussian case

(b) norm distribution in the HOG space

(c) proportion of separable points in HOG spaces of different dimensionalities

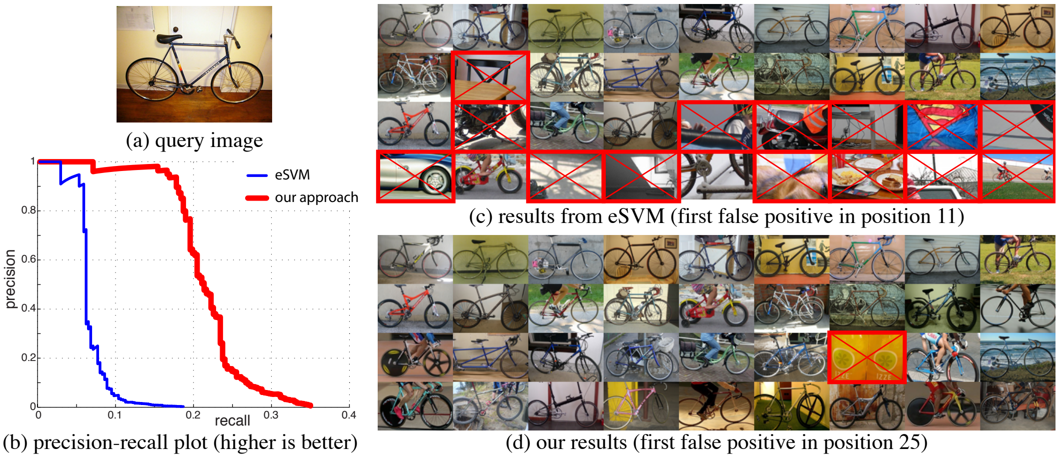
Figure 2: As the dimensionality increases, the points in a Gaussian cloud cluster farther away from the center of the cloud (a). The HOG feature vectors exhibit the same property (b). This configuration with all the points at the periphery suggests that a separating hyperplane can be found for nearly all points. To confirm this, we randomly picked 10,000 images and computed HOG descriptors of varying dimensions by changing their spatial resolution. For vectors with 125 dimensions and more, all the points can be separated (c).

Regularization. In practice, Σ may have small eigenvalues that make the inversion problematic. We address this using standard Tikhonov regularization. We seek the weights $\hat{\mathbf{w}}$ that minimize the energy $\lambda^2\|\hat{\mathbf{w}}\|^2+\|\Sigma\hat{\mathbf{w}}-(\mathbf{q}-\boldsymbol{\mu})\|^2$. In the Appendix, we show that this leads to a regularized inverse covariance matrix $\hat{\Sigma}^{-1}$ that has the same eigenvectors as Σ but with well-behaved eigenvalues $\frac{1}{\hat{s}_k} = \frac{s_k}{\lambda^2+s_k^2}$ where s_k are the eigenvalues of Σ. In practice, we set λ so that our weights best match the weights computed by the exemplar-based SVM method. Once we have this regularized matrix, we compute the normal, a.k.a. weights, as before, i.e., $\hat{\mathbf{w}} = \hat{\Sigma}^{-1}(\mathbf{q}-\boldsymbol{\mu})$.

3.1 Results on single image matching

We compare our results to the ones obtained by the exemplar-based SVM technique of Shrivastava et al. [16] that also uses a common set of negative examples for all queries. We use all images from the PASCAL VOC 2007 `trainval` set as the negative set and perform retrieval using each query image on the PASCAL VOC 2007 `test` set of images. Each query returns a sorted set of matching images, where each match is deemed correct if it overlaps by more than 0.5 with a ground-truth instances with the same category as the query. The maximum recall for our method and the baseline is low because it is computed with respect to all instances in the testing set, yet the matching is performed given a single query image. An example query bicycle image along with the top matches of the eSVM technique and the improved matches obtained by our method can be seen in Figure 3. Additional query examples along with precision-recall curves of both methods can be seen in Figure 4.

Figures 3 and 4 show that we achieve an accuracy equivalent to eSVM or even better for some categories. We hypothesize that the eSVM method may overfit the data whereas our regularization limits this problem, which explains the better accuracy in some cases. More importantly, our approach estimates the weights $\hat{\mathbf{w}}$ in 1 second on a single CPU whereas the eSVM technique takes more than 10 minutes. The computation of $\hat{\mathbf{w}}$ uses the inverse covariance matrix $\hat{\Sigma}^{-1}$ that we must compute only once, which takes approximately 4 hours on a single CPU. Finding the top matches given a single query $\mathbf{w}$ takes approximately 1 hour, because a dense sliding-window search is performed in each of the $4,965$ testing images.



(a) query image

(b) precision-recall plot (higher is better)

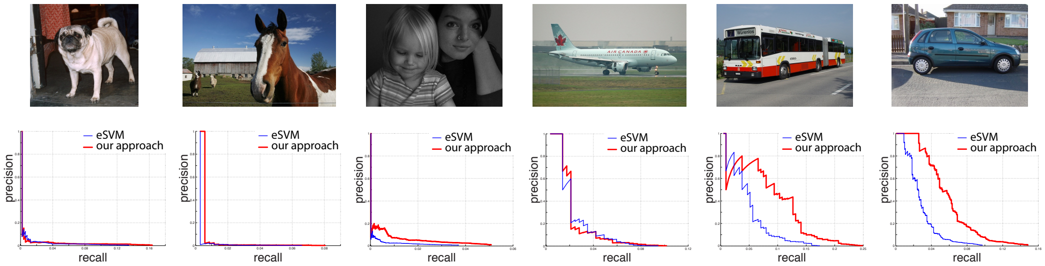
(c) results from eSVM (first false positive in position 11)

(d) our results (first false positive in position 25)

Figure 3: Comparison between eSVM and our approach with a bike query image (a). Thanks to its regularization, our method achieves a better accuracy (b), which is confirmed by visual inspection of the returned images (c,d). The maximum recall for our method and the baseline is low because it is computed with respect to all instances in the testing set, yet the matching is performed given a single query image.

3.2 Results the on PASCAL VOC2007 detection task

For comparison with the exemplar-based SVM approach [16], we evaluated our performance on the VOC2007 object detection task. All images from the `trainval` set and labeled as belonging to a given category are used as positive examples for this catergory. Images from the testing set are then analyzed. In our object detection system, each exemplar from the positive class acts as a detector



(a) a dog (b) a horse (c) a person (d) a plane (e) a bus (f) a car

Figure 4: Comparison between eSVM and our approach on several examples. Images of non-rigid objects are challenging for both methods (a,b,c) and results for rigid objects are better (d,e,f). In all cases, our approach produces results on par or better than eSVM while being orders of magnitude faster (1 second instead of more than 10 minutes).

that contributes to the overall decision of the classifier. The various outputs must then be combined to make a decision on a test window's class membership. As opposed to eSVM, our approach does not output normalized scores and natural thresholding margins. Reconciling detections from the different positive exemplars is no longer immediate. We first normalize the output scores on a per-exemplar basis then threshold the normalized scores. For each exemplar, we compute context features similar to [7, 12]. We use them to pool in association scores of overlapping detections from other exemplars. The final scores are obtained by a weighted average of these association scores. Our average performance is slightly below the exemplar-based SVM approach that uses the same negative set we used to train our covariance matrix (labeled as *'with pollution'* in Table 1). We perform better on certain classes such as $aeroplane$ or bus. Overall, we sacrifice some detection accuracy for speed.

In the current implementation the threshold level is chosen arbitrarily and is constant for all classes. This partly explains our poor performance on certain classes, for which we discard too many detections ($bird$ and $pottedplant$ for example). Our scores on the VOC2007 object detection task are summarized in Table 1 along with exemplar-based SVM and other methods.

	Our method	eSVM	eSVM	LDPM
		with pollution		
aeroplane	18.5	11.4	20.8	28.7
bicycle	38.0	39.2	48.0	51.0
bird	1.06	9.5	7.7	0.6
boat	10.5	14.3	14.3	14.5
bottle	12.7	12.4	13.1	26.5
bus	37.0	32.3	39.7	39.7
car	37.4	34.3	41.1	50.2
cat	11.4	3.5	5.2	16.3
chair	10.3	11.4	11.6	16.5
cow	11.7	19.3	18.6	16.6
dinningtable	7.0	9.6	11.1	24.5
dog	3.8	5.3	3.1	5.0
horse	29.0	38.1	44.7	45.2
motorbike	21.7	36.0	39.4	38.3
person	14.7	16.2	16.9	36.2
pottedplant	0.7	6.5	11.2	9.0
sheep	11.3	21.0	22.6	17.4
sofa	11.8	12.1	17.0	22.8
train	21.5	30.2	36.9	34.1
tvmonitor	27.9	28.1	30.0	38.4
mean	17.2	19.5	22.7	26.6

Table 1: Results on VOC2007 : *eSVM with pollution* uses the same training data as our method, *eSVM* uses a different training set for each class where no positive example pollutes the negative set. *LDPM* refers to the best performing method at the time of writing, Latent Deformable Part-based Model from P.Felzenszwalb [7]

3.3 Discussion and comparison to related work

Balanced exemplar SVM with hinge squared loss A balanced version of the exemplar SVM technique which treats the single positive and the set of negative equally seeks to find the weights $\mathbf{w}$ as:

$$\operatorname*{argmin}_{\mathbf{w}} \frac{\lambda_e}{2}||\mathbf{w}||^2 + L(\mathbf{w}^\mathsf{T}\mathbf{q}) + \frac{1}{n}\sum_{i\in\mathcal{N}} L(-\mathbf{w}^\mathsf{T}\mathbf{x}_i) \tag{3}$$

where $L(x)$ is a loss function. The original technique [14] is imbalanced and uses the hinge loss function $\max(0, 1-x)$ but it can be made more similar to our approach by considering the hinge squared loss function $\max(0, 1-x)^2$. When the problem is strongly regularized, i.e. λ_e is large, $\mathbf{w}$ becomes small, which leads to $|\mathbf{w}^\mathsf{T}\mathbf{q}| \leq 1$ and $|\mathbf{w}^\mathsf{T}\mathbf{x}_i| \leq 1\, \forall i \in \mathcal{N}$. In this configuration, Equation 3 becomes $\frac{\lambda_e}{2}||\mathbf{w}||^2 + (1-\mathbf{w}^\mathsf{T}\mathbf{q})^2 + \frac{1}{n}\sum_i(1+\mathbf{w}^\mathsf{T}\mathbf{x}_i)^2$ and accepts a closed-form solution:

$$\left(\frac{\lambda_e}{2}I + \mathbf{q}\mathbf{q}^\mathsf{T} + \boldsymbol{\mu}\boldsymbol{\mu}^\mathsf{T} + \Sigma\right)^{-1}(\mathbf{q}-\boldsymbol{\mu}) \tag{4}$$

where Σ is the same covariance matrix of the $\mathbf{x}_i$ feature vectors as we previously used. This shows that eSVM amounts to regularizing the covariance matrix by adding a constant $\lambda_e/2$ term to its eigenvalues as well as a term that depends on the query and on the mean of the space. We hypothesize that the latter term makes the effect of the regularization depend on the location in feature space, which may explain why sometime our approach produces similar results as eSVM and sometimes it performs significantly better.

LDA Our approach also shares many similarities with Linear Discriminant Analysis or Fisher's linear discriminant, which gives a closed-form equation fro the normal of the best separating hyperplane between two Gaussian distributions with parameters $(\Sigma_1, \boldsymbol{\mu}_1)$ and $(\Sigma_2, \boldsymbol{\mu}_2)$:

$$LDA(\Sigma_1, \boldsymbol{\mu}_1, \Sigma_2, \boldsymbol{\mu}_2) = (\Sigma_1 + \Sigma_2)^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) \tag{5}$$

This is analogous to our approach in which the normal is given by $\hat{\Sigma}^{-1}(\mathbf{q}-\boldsymbol{\mu})$ except that the query replaces one of the distributions.

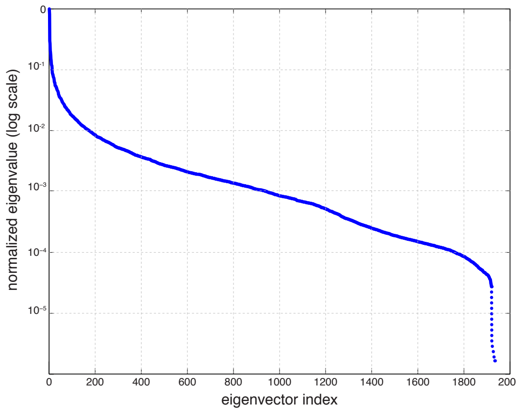


Figure 5: Eigenvalue spectrum. The value are normalized by the largest eigenvalue. For clarity's sake, the smallest eigenvalues are not represented at scale because they fall below 10^{-15} (shown with dots).

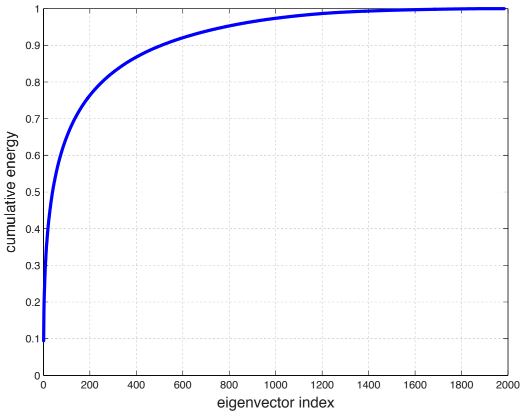


Figure 6: Eigenvalue cumulative distribution

Data size, dimensionality and degrees of freedom We believe that key to the success of our approach is that the number of training data n that we use (millions) is of the order of the square of the dimensionality d of the space (thousands). This gives us a sizable amount of data to fit the $d(d+1)/2$ coefficients of the covariance matrix. At the same time, we don't have too much data that the Gaussian assumption could be violated. A better mental picture than Fig. 1d might be a 2D Gaussian or ellipse sampled with roughly 3 points or a 3D Gaussian sampled with 6 points. It is enough data that we can fit an ellipsoid or a Gaussian, but the points are spread enough and well separable by hyperplanes.

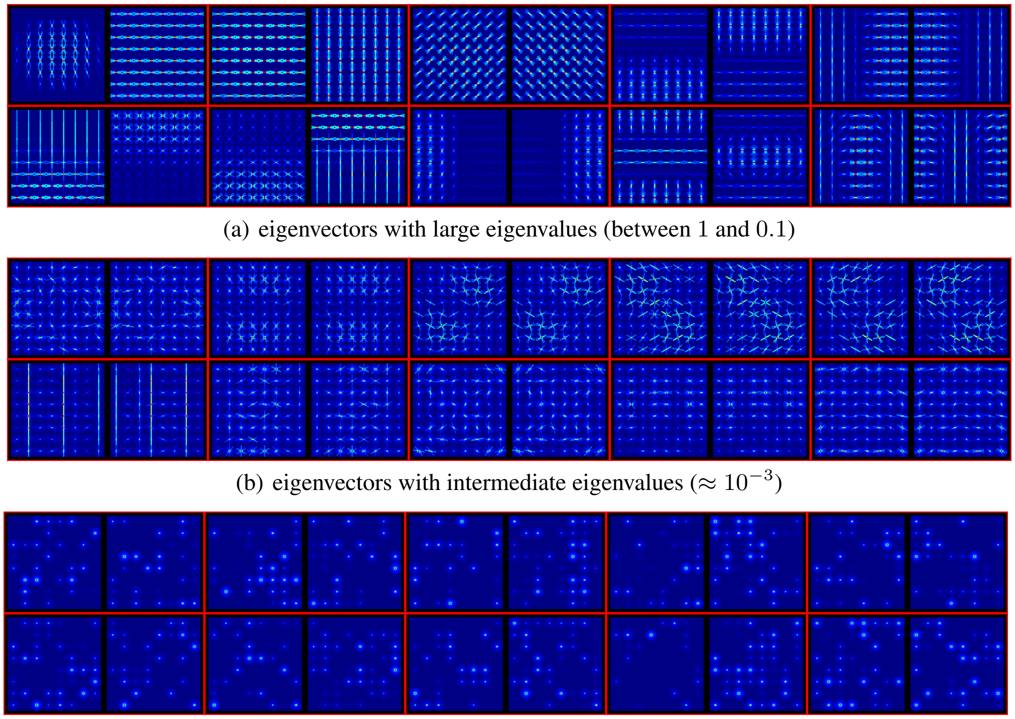
4 Whitening

The technique that results from our extension of exemplar SVM has a surprisingly simple interpretation: it amounts to a whitening of the feature space. Whitening is a standard technique in signal processing and machine learning, e.g. [6, §2.5.2], yet it is not always applied by practitioners. It is obtained by transforming the space by the inverse square root of the covariance matrix, $\Sigma^{-\frac{1}{2}}$. In Equation 2, the multiplication by the transformed feature vector is the same as the dot product in the whitened space. Whitening the space is known to "equalize" the features that is, make all feature variations more comparable, which often leads to better learning.

4.1 Eigenspace and whitened space

We study the whitening transform by considering the eigenvalues (Fig. 5 and 6) and eigenvectors (Fig. 7) of the covariance matrix. The spectrum (Fig. 5 and 6) confirms that the space is intrinsically high dimensional, with energy for a thousand dimensions. Eigenvectors with large eigenvalues (Fig. 7a) correspond to common features, which get reduced in the whitened space. They look like low frequencies in a Fourier decomposition. Small eigenvalues are characteristics that do not occur frequently and are therefore more discriminative, which is why they get amplified in the whitened space (Fig. 7b,c). However, very small eigenvalues such as in Figure 7c correspond to features that almost never occur and are considered as noise by our regularization. In the end, it is the eigenvectors with medium eigenvalues such as those in Figure 7b that play the biggest role in our similarity computation.

We can also observe the effect of the whitening transform to HOG vectors (Fig. 8). In order to be comparable to the weights visualized by Malisiewicz et al. [14], we apply the full inverse covariance Σ^{-1}, which means that the effect of whitening is applied twice, which has the benefit of making changes more obvious.

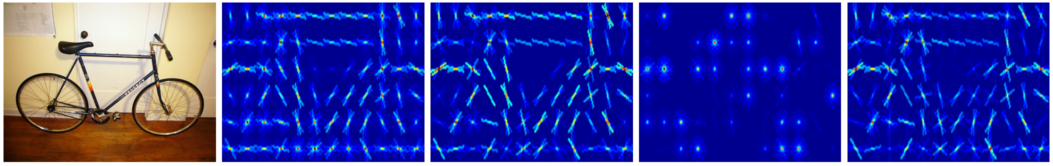


(a) eigenvectors with large eigenvalues (between 1 and 0.1)

(b) eigenvectors with intermediate eigenvalues ($\approx 10^{-3}$)

(c) eigenvectors with small eigenvalues ($\approx 10^{-5}$)

Figure 7: Eigenvectors of the covariance matrix associated with large (a), intermediate (b), and small (c) eigenvalues. The positive and negative parts are shown for each vector.



(a) input (b) unweighted HoG descriptor (c) eSVM weights (d) unregularized weights (e) regularized weights

Figure 8: Effect of feature whitening.

4.2 Discussion and related work

Distance. Given a whitened feature space, one can compute nearest neighbors based on a variety of metrics. Th Euclidean distance is always a tempting option, but as Figure 2 demonstrated, distances in high-dimensional spaces tend to be clumped, making it a poor option. We tried to use the Euclidean distance but even after whitening, it performed barely above chance.

In contrast, the dot product measure of similarity performs well, as shown in Figure 3. We also made tests with feature vectors normalized in the whitened space. In this case, both the dot product and Euclidean distance performed equally well, which is not surprising since they are linked by: $\|\mathbf{x}-\mathbf{y}\|^2 = \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - 2\,\mathbf{x}\cdot\mathbf{y}$.

Link to Latent Semantic Analysis. Latent Semantic Analysis, e.g. [11], is used for many applications in document analysis and retrieval. It characterizes the content of text passages (e.g. paragraphs) as vectors of word occurrences. SVD is then used to capture co-occurences. Passages can be compared by taking the resulting dot product. The use of SVD is similar to our whitening, and they also use a dot product as measure of similarity.

Uniqueness and saliency. This allows us to revisit the notion of image "uniqueness" proposed by Shrivastava et al. [16] and Boiman and Irani [3] and why eSVM relates to image saliency [16], which seeks to characterize where eyes look in images. For this, we turn to Rosenholtz's work [15], which predicts pop out phenomena such as the salience of a square in the middle of an array of circle. She argues that, if the distribution of visual features such as contrast and color is modeled as a Gaussian, elements such as the square are outliers. This can be measured by taking the Mahalanobis distance induced by the covariance of all the observed features, which is the same as the Euclidean distance in the whitened space.

If we extend this notion to the space of all images and consider features that are HOG descriptors, then the whitened HOG descriptors tells us what is special about an image, de-emphasizing characteristics that are common to many images.

5 Conclusions

After a winding journey that started from exemplar-based SVM, we introduced simple approach to image matching that transforms the feature vectors according to the inverse covariance matrix and uses the dot product as a measure of similarity. It corresponds to feature whitening and linear discriminant analysis, and has links to image saliency. Despite its simplicity, it performs similarly to state of the art techniques, e.g. [14, 16] and is orders of magnitude faster.

A Tikhonov regularization of the covariance matrix

The covariance matrix Σ may not be invertible and/or may have very small eigenvalues that makes the inversion unstable. We address this by regularizing the inversion and seek weights $\hat{\mathbf{w}}$ that satisfy $\Sigma\hat{\mathbf{w}} = \mathbf{q} - \boldsymbol{\mu}$ while being small. We model this as the minimization of the objective function $\lambda^2\|\hat{\mathbf{w}}\|^2 + \|\Sigma\hat{\mathbf{w}} - (\mathbf{q} - \boldsymbol{\mu})\|^2$. The derivative with respect to $\hat{\mathbf{w}}$ is zero at the solution, which gives us $(\lambda^2 I + \Sigma^2)\hat{\mathbf{w}} = \Sigma(\mathbf{q} - \boldsymbol{\mu})$ using the symmetry of Σ. We decompose Σ into its eigenvalues s_k and its eigenvectors $\mathbf{u}_k$, and express the weights in this basis, $\hat{w}_k = \mathbf{u}_k^\intercal \hat{\mathbf{w}}$. This leads to $(\lambda^2 + s_k^2)\hat{w}_k = s_k(q_k - \mu_k)$ and finally: $\hat{w}_k = \frac{s_k}{\lambda^2 + s_k^2}(q_k - \mu_k)$.

References

[1] R. E. Bellman. *Adaptive control process*. Princeton University Press, 1961.

[2] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[3] O. Boiman and M. Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision*, 74(1), 2007.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Computer Vision and Pattern Recognition*, 2005.

[5] D. L. Donoho. High-dimensional data analysis : The curses and blessings of dimensionality. *Statistics*, 2000.

[6] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley and Sons, 2001.

[7] P. F. Felzenszwalb, R. B. Girschick, D. McCallester, and D. Ramanan. Object detection with discriminatively trained part based models. *Pattern Analysis and Machine Intelligence*, 2010.

[8] A. Frome, Y. Singer, and J. Malik. Image retrieval and classification using local distance functions. In *Neural Information Processing Systems*, 2006.

[9] B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In *European Conference on Computer Vision (ECCV)*, 2012.

[10] T. Hastie and R. Tibshirani. Discriminant adaptive nearest neighbor classification. *Pattern Analysis and Machine Intelligence*, 18(6), 1996.

[11] T. K. Landauer, P. W. Foltz, and D. Laham. Introduction to latent semantic analysis. *Discource Processes*, 25, 1998.

[12] T. Malisiewicz. Exemplar-based representations for object detection, association and beyond. *Carnegie Mellon University PhD Thesis*, 2011.

[13] T. Malisiewicz and A. A. Efros. Recognition by association via learning per-exemplar distances. In *Computer Vision and Pattern Recognition*, 2008.

[14] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-SVMs for object detection and beyond. *International Conference on Computer Vision*, 2011.

[15] R. Rosenholtz. A simple saliency model predicts a number of motion popout phenomena. *Vision Research*, 39, 1999.

[16] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. *ACM Transaction of Graphics*, 30(6), 2011.