One Shot Detection with *Laplacian Object* and Fast Matrix Cosine Similarity

Sujoy Kumar Biswas, *Student Member, IEEE,* Peyman Milanfar, *Fellow, IEEE*

***Abstract*—One shot, generic object detection involves searching for a single query object in a larger target image. Relevant approaches have benefited from features that typically model the local similarity patterns. In this paper, we combine local similarity (encoded by local descriptors) with a global context (i.e., a graph structure) of pairwise affinities among the local descriptors, embedding the query descriptors into a low dimensional but discriminatory subspace. Unlike principal components that preserve global structure of feature space, we actually seek a linear approximation to the Laplacian eigenmap that permits us a locality preserving embedding of high dimensional region descriptors. Our second contribution is an accelerated but exact computation of matrix cosine similarity as the decision rule for detection, obviating the computationally expensive sliding window search. We leverage the power of Fourier transform combined with integral image to achieve superior runtime efficiency that allows us to test multiple hypotheses (for pose estimation) within a reasonably short time. Our approach to one shot detection is training-free, and experiments on the standard data sets confirm the efficacy of our model. Besides, low computation cost of the proposed (codebook-free) object detector facilitates rather straightforward query detection in large data sets including movie videos.**

***Index Terms*—One shot object detection, Graph based dimensionality reduction, Fourier transform, Fast Detection**

I. Introduction

Recent research in visual recognition [1] has attracted interest in the study of the following two big questions — i) how to make generalizations for solving large scale visual recognition problem [2], [3], and ii) how to encode image attributes robustly to represent fine-grained distinction/similarity [4]. The former question aims to understand visual content at large scale, generalizing patterns from millions of images for thousands of class labels. In contrast, fine-grained visual recognition engages in fine-scale distinction among categories which are both visually and semantically similar (e.g., identifying each of the 100 *models* from 10,000 aircraft images [5]). It naturally follows that in the study of fine-scale visual similarity, stronger feature encoding and robust matching strategies require careful attention. In this paper, we focus on a particular variety of the fine-grained visual recognition where the objective is to detect visual similarities (Fig. 1) across images without the involvement of extensive (or any) training. In general, the one shot, generic object detection approaches take a single query image as input, and the dominant object present in the query image is detected in a bigger target image. Recent studies [6], [7], [8] have shown that exemplar-based, training-free detection can work with laudable success, sometimes very close to training-based approaches. But more importantly, such detection strategies provide interesting insights into developing newer and better features along with provably useful matching strategies.

Query objects typically appear in target images with wide variations both geometric as well as optical. Geometric variations can include severe changes in scale and orientation (pose) of the query, whereas optical variations may result from differences in lighting, resolution and noise level. Besides, presence of clutter and not having enough out-of-class examples make the detection task prone to false alarms. We rely on a sophisticated embedding technique to obtain a compact but discriminative set of features to deal with such challenges. Moreover, we employ efficient computational accelerations that lead to exact evaluation of decision rules for detection but in several order of magnitude faster than the traditional sliding window based detection schemes. Such faster computation of decision rule allows us to test a greater number of hypotheses within a relatively short time for estimating pose as well.

A. Overview, Past Work, and Proposed Contributions

We denote the query and target images by Q and T respectively, and compute high dimensional descriptors densely over both query and target, storing them in descriptor matrices $\mathbf{H}_Q$ and $\mathbf{H}_T$, respectively. The dense computations make the descriptors highly informative but also redundant. Hence, to facilitate fast, efficient and effective detection we extract compact but salient features $\mathbf{F}_Q$ and $\mathbf{F}_T$ from the high dimensional descriptors. Since, T is bigger than query Q, we sweep the query window over target, and comparing features $\mathbf{F}_Q$ and $\mathbf{F}_{T_i}$ (extracted from i-th position of the sliding window over T) we estimate the likelihood of the presence of Q in T (Fig. 1).

Relevant approaches have benefitted from descriptors that typically model the local similarity patterns. This is because the image is often replete with self-similar patterns. Motivated by this observation, Shechtman *et al.* [9] proposed a pattern matching scheme based on *self-similarity*, the premise of which is that local internal layout of self-similar pixels are shared by visually similar images. Since its introduction, self-similarity based models and approaches have found several applications in subsequent research in computer vision ranging from detection to object and sketch retrieval [10], [11], [12]. The local self-similarity [9] was defined as a function of simple sum of square difference (SSD) between a center image patch and neighboring image patch. In [6], Seo *et al.*, modeled local geometric layout with *Locally Adaptive Regression Kernel* (LARK) [13] descriptors, and projected them on principal components to extract compact and discriminative features. We

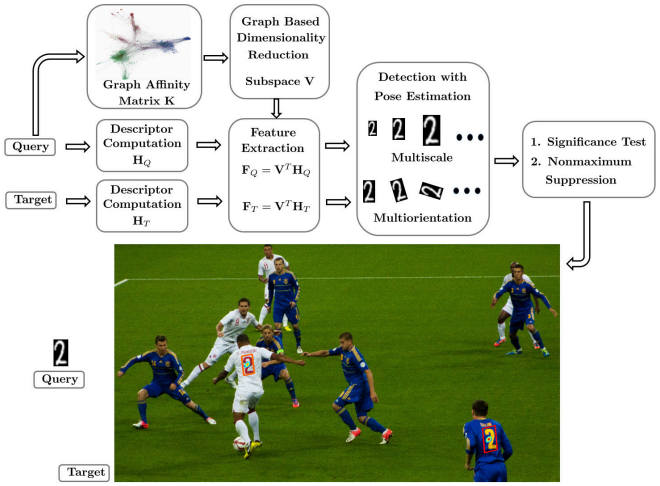


Fig. 1. Overview of our one shot detection scheme: we aim to detect a given query [15] (e.g., symbol, face, human pose, car, flower) appearing in a visually similar manner in a bigger target image [1]

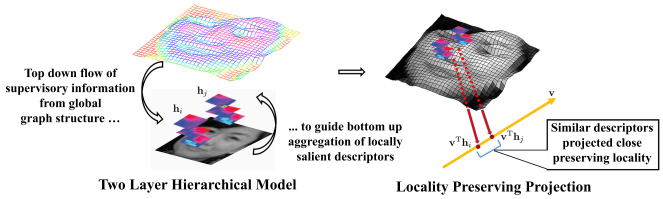


Fig. 2. *Laplacian Object*: computing a query subspace that preserves intrinsic image geometry — on left, the proposed two-layer hierarchical model is shown where top layer of global context (in the form of an affinity graph) guides the bottom up aggregation of local information from low level descriptors. On right, locality preserving projection [17] with the graph Laplacian is used as a mathematical framework to represent the two-layer hierarchy.

endorse their view but note that while local feature encoding is important, the global context (as in [14]) can not be ignored. We argue that while projecting descriptors on a discriminatory subspace, as in [6], it is imperative to consider relative position of descriptors so as to preserve the intrinsic geometry of image pattern (PCA does not consider spatial location of descriptors).

One common aspect that emerges from such competing methods is the emphasis on the role of i) local geometry in building salient features, and ii) global context that encapsulates the spatial information of descriptors. In what follows, we propose a two-layer hierarchical model as shown in Fig. 2 for combining local geometry with global context toward obtaining salient features. The top layer passes global information to guide the bottom up aggregation of low level visual cues. Our ultimate goal is to estimate a low dimensional subspace where the high dimensional region descriptors could be projected with their local image geometry intact.

Sliding window search for object detection is tedious and computationally demanding. The second contribution of this paper involves accelerating the computation of matrix cosine similarity (a generalization of cosine similarity for matrix features $\mathbf{F}_Q$ and $\mathbf{F}_{T_i}$) as the decision rule to detect objects. Since at the heart of matrix cosine similarity (MCS) there lies a correlation computation, motivated by Dubout *et al.* [16] we employ the use of *Discrete Fourier Transform* (DFT) for evaluating correlation efficiently in frequency domain. MCS has also a normalization factor that is not amenable to spectral techniques. The trick to efficient computation of the normalization factor lies in a precomputed area sum table (integral image) of target feature vectors' L_2 norm. The integral image allows constant time retrieval of the normalization factor independent of the size of the sliding window.

The rest of the manuscript is organized as follows. Section II describes the foundation of the graph based dimensionality reduction technique for the proposed hierarchical model, Section III introduces a unifying view of local descriptors (bottom layer) and and the graph structure (top layer), Section IV presents the detection framework and the proposed faster detection scheme, Section V reports experimental results followed by relevant discussion, and we draw conclusions in Section VI.

[1]The original picture (available online, 7th September, 2014: https://www.flickr.com/photos/tdd/8504835638/) used (strictly for academic purpose) is owned by Tomasz Dunn, and licensed under Creative Commons Attribution 2.0 Generic (CC BY 2.0) for free use and modification with attribution.

II. *Laplacian Object*: A Framework for Locally Salient Feature Computation

We begin the model description by assuming $m \times n$ sized gray-scale Q and $M \times N$ sized T. We visualize a gray-scale image as the parameterized image surface $\mathcal{S}(\mathbf{x}_i) = \{\mathbf{x}_i, z(\mathbf{x}_i)\}$, where $\mathbf{x}_i$ denotes the 2D coordinate vector $\mathbf{x}_i = [x_{i_1}, x_{i_2}]^{\mathrm{T}}$, having intensity $z(\mathbf{x}_i)$. We compute local image descriptors (e.g., SIFT [18], LARK [19]) densely at every pixel, that makes the number of descriptors from Q as mn, and from T as MN. The descriptor at location $\mathbf{x}_i$ is denoted as a l-dimensional vector $\mathbf{h}_i \in \mathbb{R}^l$. The descriptor vectors $\mathbf{h}_{Qi}$ for query, and $\mathbf{h}_{Ti}$ for target, are stacked column wise to define the descriptor matrix for query as $\mathbf{H}_Q = [\mathbf{h}_{Q1}, \mathbf{h}_{Q2}, \ldots, \mathbf{h}_{Qmn}] \in \mathbb{R}^{l \times (mn)}$, and the same for target as $\mathbf{H_T} = [\mathbf{h}_{T1}, \mathbf{h}_{T2}, \ldots, \mathbf{h}_{TMN}] \in \mathbb{R}^{l \times (MN)}$. To distill the redundancy resulting from dense computation of descriptors we embed $\mathbf{H}_Q$ in a global graph structure, represented by an affinity matrix $\mathbf{K}$, that takes into account the spatial relationship among the descriptors. Our goal is to estimate a low dimensional but discriminatory subspace $\mathbf{v}$ from Q such that the query descriptors $\mathbf{h}_{Qi}$, when projected on $\mathbf{v}$, respect the local geometric pattern. In other words, if $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ are closely spaced over the image manifold $\mathcal{S}$ then their projections $\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qi}$ and $\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qj}$ on a subspace $\mathbf{v}$ should be close as well (Fig. 2). The theory of locality preserving projection (LPP) [20] ensures this criterion by minimizing the following objective function —

$$J_{\mathrm{LPP}} = \frac{1}{2}\sum_{ij}(\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qi} - \mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qj})^2\mathbf{K}_{ij}. \quad (1)$$

The objective function J_{LPP} with our proposed affinity measure $\mathbf{K}_{ij}$ incurs heavy penalty if *neighboring* descriptors $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ are mapped *far apart*. Each element of graph affinity matrix $\mathbf{K}$, denoted as $\mathbf{K}_{ij}$, indicates the pairwise affinity

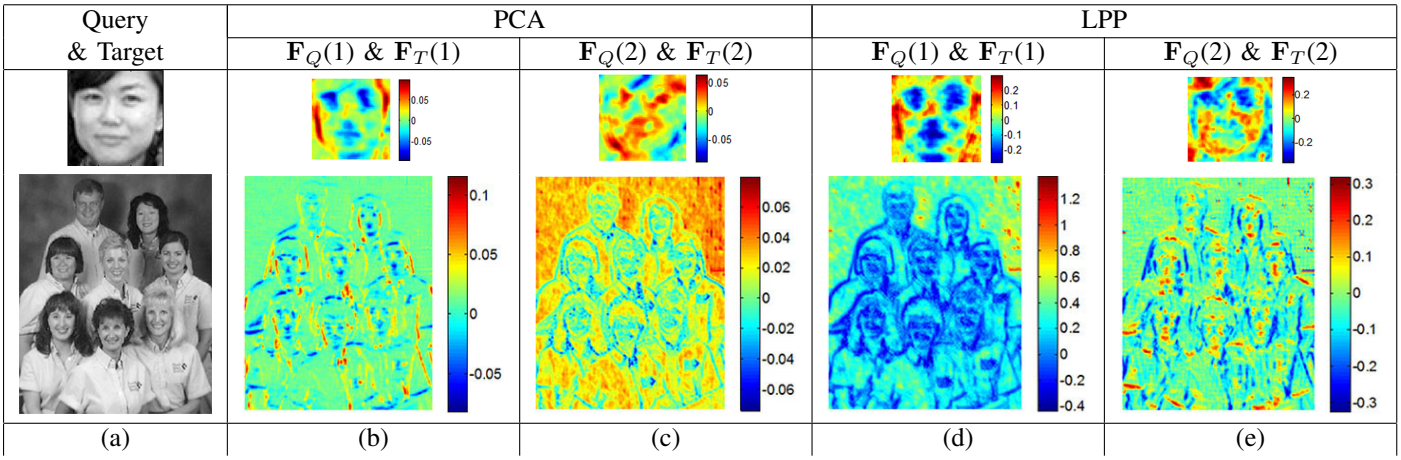


Fig. 3. Salient features shown after dimensionality reduction of LARK descriptors: (a) query & target images, (b)-(c) salient query (target) features $\mathbf{F}_Q$ ($\mathbf{F}_T$) learnt by projecting descriptors $\mathbf{H}_Q$ ($\mathbf{H}_T$) along two dominant principal components, (d)-(e) same LARK descriptors projected along two dominant eigenvectors of LPP (one can notice finer local details in these features)

between descriptors $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ of the query Q. Upon simplification, (1) leads to the following penalty (for details refer to [20]):

$$J_{\mathrm{LPP}} = \mathbf{v}^{\mathrm{T}}\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^{\mathrm{T}}\mathbf{v}. \tag{2}$$

Defining the diagonal matrix $\mathbf{D}_{ii} = \sum_j \mathbf{K}_{ij}$, the matrix $\mathbf{L} = \mathbf{D} - \mathbf{K}$ is known as the graph Laplacian. Recent research shows success of graph Laplacian in effective exploration of local patterns in massive graphical networks [21], [22]. Here we have studied the related Laplacian eigenmap [23], [24] to embed the informative but redundant descriptors into a low dimensional but salient feature space (Fig. 2).

To prevent abnormally high values of $\mathbf{D}_{ii}$ (which means unusually greater "importance" to descriptor $\mathbf{h}_i$) in (2), the constraint $\mathbf{v}^{\mathrm{T}}\mathbf{H_Q}\mathbf{D}\mathbf{H_Q}^{\mathrm{T}}\mathbf{v} = 1$ is imposed on $\mathbf{D}$. Minimizing J_{LPP} with respect to the aforementioned constraint we obtain the following optimization problem:

$$\min_{\mathbf{v}} \mathbf{v}^{\mathrm{T}}\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^{\mathrm{T}}\mathbf{v} \text{ subject to } \mathbf{v}^{\mathrm{T}}\mathbf{H}_Q\mathbf{D}\mathbf{H}_Q^{\mathrm{T}}\mathbf{v} = 1. \tag{3}$$

The projection vector $\mathbf{v}$ that minimizes the above is given by the minimum eigenvalue solution to the following generalized eigenvalue problem:

$$\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^{\mathrm{T}}\mathbf{v} = \lambda\mathbf{H}_Q\mathbf{D}\mathbf{H}_Q^{\mathrm{T}}\mathbf{v}. \tag{4}$$

The desired set of eigenvectors which builds our low dimensional LPP subspace comprises the trailing d eigenvectors computed as a solution of (4). We collect the set of d eigenvectors as columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d] \in \mathbb{R}^{l\times d}$. Since the descriptors are densely computed they typically lie on a lower dimensional manifold. As a consequence, we can expect d to be quite small in comparison to the dimension l of the descriptors. In practice, d is selected to be a small integer, and it turns out that this small set of eigenvectors is good enough to discriminate the query from the background clutter. The descriptor matrices $\mathbf{H}_Q$ and $\mathbf{H}_T$, when projected on $\mathbf{V}$, lead to salient features that preserve locality as guaranteed by the objective function (1). The locally salient features $\mathbf{F}_Q$ and $\mathbf{F}_T$, for query Q and target T respectively, are defined by the following equations:

$$\mathbf{F_Q} = \mathbf{V}^{\mathrm{T}}\mathbf{H}_Q \in \mathbb{R}^{d\times(mn)}; \mathbf{F_T} = \mathbf{V}^{\mathrm{T}}\mathbf{H}_T \in \mathbb{R}^{d\times(MN)}. \tag{5}$$

The salient query (target) features $\mathbf{F}_Q$ ($\mathbf{F}_T$) learnt with PCA and LPP are shown in Fig. 3. The results in the figure(s) demonstrate that LPP is able to preserve greater amount of details in the projected features than PCA. During detection the detailed contour and inherent spatial geometry captured in LPP result in better localization. The reason why LPP features inculcate more information compared to PCA features [6] lies in the construction of respective objective functions as explained below.

Descriptors $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qi}$ typically encode geometric information in various channels, but if they **share** similar local geometry they would likely have a high pairwise similarity term $\mathbf{K}_{ij}$. Consequently, a high $\mathbf{K}_{ij}$ would penalize the cost function in case $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qi}$ are projected far apart. The pairwise similarity term ensures that the local continuity of the fine edge structure would be preserved on the projected subspace.

PCA does not care which descriptor comes from where – it retains the global geometric structure of the data (as evident from the mean term $\overline{h}$ below) without providing any room for preserving local details. It maximizing the following objective function,

$$J_{\mathrm{PCA}} = \sum_i (\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qi} - \mathbf{v}^{\mathrm{T}}\overline{\mathbf{h}})^2 \tag{6}$$

The graph encoded by $\mathbf{K}_{ij}$ in (1) provides further insights into the working principles of LPP and PCA. We denote the total number of pixels mn in query by n_q. Suppose we connect $\mathbf{x}_i$ with all other pixels $\mathbf{x}_j$ of the query image obtaining a complete graph with constant weights $\mathbf{K}_{ij} = \frac{1}{n_q^2}, \forall \mathbf{x}_i, \mathbf{x}_j$. Then $\mathbf{L} = \mathbf{D} - \mathbf{K} = \frac{1}{n_q^2}\mathbf{I} - \frac{1}{n_q^2}\mathbf{e}\mathbf{e}^{\mathrm{T}}$, where $\mathbf{e}$ is a vector of all ones. Under this graph construction, and denoting mean

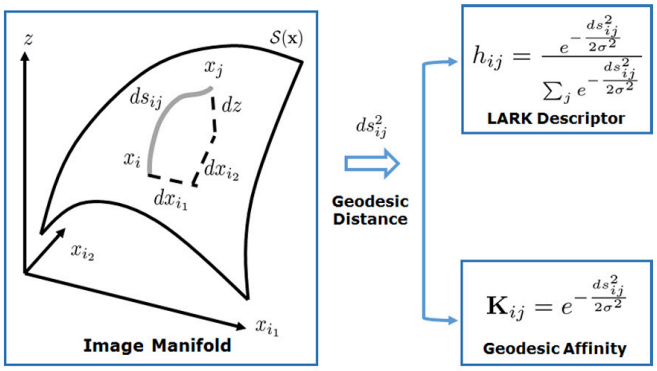


Fig. 4. Unifying geodesic framework: the geodesic distance (ds_{ij}) between the points $\mathbf{x}_i$ and $\mathbf{x}_j$ on the image manifold $\mathcal{S}(\mathbf{x})$ is used to derive both the LARK descriptors and affinities on the right

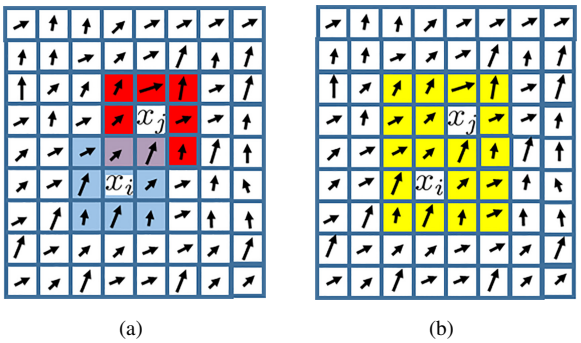


Fig. 5. Estimation of covariance matrix $\mathbf{C}$ from local gradients (shown with black arrows): (a) For LARK descriptors we estimate $\mathbf{C}_{\Omega_i}$ from (13) using the support patch Ω_i corresponding to $\mathbf{x}_i$ as shown in (blue) color. Note, Ω_j (in red) corresponding to $\mathbf{x}_j$ is different from Ω_i. To make $\mathbf{K}_{ij}$ symmetric (b) shows the rule adopted for defining a common support for $\mathbf{x}_i$ and $\mathbf{x}_j$ using the patch Ω_{ij} (shown in yellow) over which $\mathbf{C}_{\Omega_{ij}}$ is estimated (15).

$\overline{h} = \frac{1}{n_q}\sum_i \mathbf{h}_{Qi}$ we get,

$$\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^T = \frac{1}{n_q}\mathbf{H}_Q(\mathbf{I} - \frac{1}{n_q}\mathbf{e}\mathbf{e}^{\mathrm{T}})\mathbf{H}_Q^{\mathrm{T}}, \tag{7}$$

$$= \frac{1}{n_q}\mathbf{H}_Q\mathbf{H}_Q^{\mathrm{T}} - \frac{1}{n_q^2}(\mathbf{H}_Q\mathbf{e})(\mathbf{H}_Q\mathbf{e})^{\mathrm{T}}, \tag{8}$$

$$= \frac{1}{n_q}\sum_i \mathbf{h}_{Qi}\mathbf{h}_{Qi}^{\mathrm{T}} - \frac{1}{n_q^2}(n\bar{\mathbf{h}})(n\bar{\mathbf{h}})^{\mathrm{T}}, \tag{9}$$

$$= \frac{1}{n_q}\sum_i (\mathbf{h}_{Qi} - \bar{\mathbf{h}})(\mathbf{h}_{Qi} - \bar{\mathbf{h}})^{\mathrm{T}}. \tag{10}$$

This happens to be the covariance matrix of the data set that is used in PCA (J_{PCA} upon simplification boils down to $J_{\mathrm{PCA}} = \mathbf{v}^{\mathrm{T}}\left[\sum_i (\mathbf{h}_{Qi} - \bar{\mathbf{h}})(\mathbf{h}_{Qi} - \bar{\mathbf{h}})^{\mathrm{T}}\right]\mathbf{v}$). The analysis above (and also in [20]) suggests when we care about global structure of LARK descriptor space we connect each descriptor location (i.e., each pixel) to all others in the graph construction, and project the descriptors along the direction of maximal variance. When we seek to preserve local information in reduced dimension we connect each pixel to its immediate neighborhood, and project the descriptors along the direction that minimizes local variation.

A fact of theoretical interest is the subtle difference between Laplacian eigenmap [23] and LPP [20]. The former provides us a non-linear manifold, and a (linear) projection operation like (5) does not hold true in case of Laplacian eigenmap. LPP, as described in [20], can be seen as a linear approximation to the non-linear Laplacian eigenmap allowing us to project the query and target features as shown in (5).

III. Local Descriptors & Affinity Matrix From a Unifying Geodesic Perspective

The proposed detection framework is general enough to use with any local descriptor (e.g., SIFT [18], HOG[25]). However, we advocate LARK descriptor because it is specifically designed for one shot object detection [6], [26]. In fact, the geodesic interpretation behind LARK descriptors introduced in [26] motivates us to present an unifying geometric perspective to connect the definitions of LARK descriptor $\mathbf{h}_i$ and the graph affinity $\mathbf{K}_{ij}$.

The local geodesic distance (Fig. 4) between the two neighboring descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$ on the image manifold $\mathcal{S}(\mathbf{x}_i)$ is approximated [26] by the differential arc length ds_{ij} as follows:

$$ds_{ij}^2 = dx_{i_1}^2 + dx_{i_2}^2 + dz^2 \approx \Delta x_{ij}^{\mathrm{T}}\mathbf{C}_i\Delta x_{ij}. \tag{11}$$

The approximation involves the following discretizations: $dx_{i_1} \approx \Delta x_{i_1 j_1} = x_{j_1} - x_{i_1}$, and $dx_{i_2} \approx \Delta x_{i_2 j_2} = x_{j_2} - x_{i_2}$ (i.e., $\Delta x_{i_1 j_1}$ and $\Delta x_{i_2 j_2}$ representing displacements along the two image-axes in Fig. 4). Also, we assume $\Delta x_{ij} = [\Delta x_{i_1 j_1} \quad \Delta x_{i_2 j_2}]^{\mathrm{T}}$, and the matrix $\mathbf{C}_i$ denotes the local gradient covariance matrix (also called as steering matrix in [13]) computed at $\mathbf{x}_i$.

A. Computation of LARK descriptors

The descriptor $\mathbf{h}_i$ denotes a multidimensional vector $\mathbf{h}_i = (h_{i1}, h_{i2}, \ldots, h_{ij}, \ldots, h_{p^2})$ computed at pixel $\mathbf{x}_i$ over a $p \times p$ local window. We define the general term h_{ij} as a measure of similarity between two descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$ as follows:

$$h_{ij} = \frac{e^{-\frac{ds_{ij}^2}{2\sigma^2}}}{\sum_{j=1}^{p^2} e^{-\frac{ds_{ij}^2}{2\sigma^2}}}, \quad j = 1, 2, \ldots, p^2, \tag{12}$$

where ds_{ij} is approximated as in (11). The normalization in the denominator is carried out by summing the local geodesic similarities over all the neighbors of $\mathbf{x}_i$ in its $p \times p$ local neighborhood. LARK descriptors when normalized to a unit vector become robust to illumination changes.

Straightforward computation of $\mathbf{C}_i$ in (11) based on raw image gradient at a single pixel may be too noisy. Therefore, to estimate $\mathbf{C}_i$ in a robust fashion, first we compute the derivatives of the image signal $z(\mathbf{x}_i)$ over a patch Ω_i of pixels centered at pixel $\mathbf{x}_i$ (Fig. 5(a)) and we denote such local gradient covariance matrix as $\mathbf{C}_{\Omega_i}$. This accumulation of first derivatives guards against the undesirable effect of noise and perturbations. Secondly, we further smooth the signal manifold, to strictly focus on the dominant pattern of local

texture, by computing $\mathbf{C}_{\Omega_i}$ in a stable way that includes eigen-decomposition. Combining these two steps we write the final expression of $\mathbf{C}_{\Omega_i}$ as follows:

$$\begin{aligned}\mathbf{C}_{\Omega_i} &= \sum_{m\in\Omega_i}\begin{pmatrix}\frac{\Delta z(m)}{\Delta x_{i_1}}^2 & \frac{\Delta z(m)}{\Delta x_{i_1}}\cdot\frac{\Delta z(m)}{\Delta x_{i_2}}\\ \frac{\Delta z(m)}{\Delta x_{i_1}}\cdot\frac{\Delta z(m)}{\Delta x_{i_2}} & \frac{\Delta z(m)}{\Delta x_{i_2}}^2\end{pmatrix},\\ &= \nu_1\mathbf{u_1}\mathbf{u_1}^{\mathrm{T}}+\nu_2\mathbf{u_2}\mathbf{u_2}^{\mathrm{T}},\\ &= (\sqrt{\nu_1\nu_2}+\varepsilon)^{\theta}\cdot\\ &\quad(\frac{\sqrt{\nu_1}+\tau}{\sqrt{\nu_2}+\tau}\mathbf{u_1}\mathbf{u_1}^{\mathrm{T}}+\frac{\sqrt{\nu_1}+\tau}{\sqrt{\nu_2}+\tau}\mathbf{u_2}\mathbf{u_2}^{\mathrm{T}}),\end{aligned}\quad(13)$$

where, ν_1 and ν_2 are eigenvalues of $\mathbf{C}_{\Omega_i}$ corresponding to eigenvectors $\mathbf{u}_1$ and $\mathbf{u}_2$, respectively. Also in the derivation above, $\varepsilon, \tau, \theta$ are regularization parameters to avoid numerical instabilities and kept constant throughout all the experiments in this paper at $10^{-7}, 1$ and 0.1 respectively.

B. Building the Graph Laplacian with Geodesic Affinities

Next, we build a graph structure from Q with descriptors representing the graph nodes. The edges in the graph denote affinities between neighboring descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$ as follows,

$$\mathbf{K}_{ij}=\begin{cases}e^{-\frac{ds_{ij}^2}{2\sigma^2}} & \text{when } \|\mathbf{x}_i-\mathbf{x}_j\|_2\le\gamma,\\ 0 & \text{otherwise},\end{cases}\quad(14)$$

where σ is a smoothing parameter (kept same in (12) for LARK descriptor), and γ a radius within which we limit the affinity computation. The choice of γ is not too critical as long as it covers decent neighborhood size (typically 3 to 5 pixel radius). Setting γ too high increases the computational burden, and may involve derogatory confluence of too many and irrelevant neighborhood information. In fact, as also observed in [20], too much aggregation of information collected over a bigger neighborhood may invariably affect LPP's embedding performance.

Unfortunately, $\mathbf{K}_{ij}$ defined above is non symmetric. However, the derivation of LPP subspace in (2) assumes a symmetric affinity matrix $\mathbf{K}$. To understand why $\mathbf{K}_{ij}$ is non symmetric we note computing $ds_{ij}=\Delta x_{ij}^{\mathrm{T}}\mathbf{C}_{\Omega_i}\Delta x_{ij}$ following the definition of $\mathbf{C}_{\Omega_i}$ in (11) makes $ds_{ij}\neq ds_{ji}$. This is because the support Ω_i of $\mathbf{C}_{\Omega_i}$ is centered at $\mathbf{x}_i$ (Fig. 5(a)), and similarly, support Ω_j of $\mathbf{C}_{\Omega_j}$ is centered at $\mathbf{x}_j$, hence, $\mathbf{C}_{\Omega_i}\neq\mathbf{C}_{\Omega_j}$. Therefore, to ensure $\mathbf{K}_{ij}$ to be symmetric we make the supports of $\mathbf{C}_{\Omega_i}$ and $\mathbf{C}_{\Omega_j}$ common as shown by a circumscribing rectangle in Fig. 5(b). The common support is denoted by Ω_{ij}, and we write the corresponding gradient covariance matrix as $\mathbf{C}_{\Omega_{ij}}$. It follows directly that $ds_{ij}=ds_{ji}=\Delta x_{ij}^{\mathrm{T}}\mathbf{C}_{\Omega_{ij}}\Delta x_{ij}$, and the final expression of affinity becomes the following:

$$\mathbf{K}_{ij}=\begin{cases}e^{-\frac{\Delta x_{ij}^{\mathrm{T}}\mathbf{C}_{\Omega_{ij}}\Delta x_{ij}}{2\sigma^2}} & \text{when } \|\mathbf{x}_i-\mathbf{x}_j\|_2\le\gamma,\\ 0 & \text{otherwise}.\end{cases}\quad(15)$$

Note, to ensure symmetry a straightforward averaging ($\mathbf{K}_{ij}=\mathbf{K}_{ji}=\frac{\mathbf{K}_{ij}+\mathbf{K}_{ji}}{2}$) does not work well in practice because such average oversmooths the dominant structure pattern over the image manifold.

To summarize, since each LARK channel captures a specific orientation pattern, different edge orientations manifest themselves in different LARK channels along with relative signal strength of the edges (in terms of ν_1, ν_2 in (13)). Flat regions receive low values in all the channels, but edge structures show up as high values in appropriate channel depending on the orientation. Getting all the directional information of 81 channels (when $p=9$) in 5 or 6 low dimension is difficult: when PCA does this job it tends to show the fine structures like eyes, nose, or mouth of faces (or contour of parts in case of generic objects) as blobs in high contrast regions, and completely misses the relatively faint parts in low contrast region (Fig. 3). In contrast, LPP is able to retain the delicate image geometry relatively better. The reason is by virtue of pairwise similarity LPP keeps the projected descriptors close, if similar, thereby maintaining local continuity of fine details. Viewed from an alternative perspective, LPP projects the high dimensional LARK channels along a direction that minimizes the weighted local variance following a least square framework (1), preserving geometric details in lower dimension.

IV. Detection Framework and Faster Query Search

A. Detection Framework, FDR Control & Pose Estimation

The traditional sliding window based detection sweeps the query window over T, and at each position $\mathbf{x}_i$ (center of sliding window) in T the MCS decision rule [6] is computed as follows:

$$\rho_i=\rho(\mathbf{F}_Q,\mathbf{F}_{T_i})=\mathrm{trace}(\frac{\mathbf{F}_Q^{\mathrm{T}}\mathbf{F}_{T_i}}{\|\mathbf{F}_Q\|_F\|\mathbf{F}_{T_i}\|_F})\in[-1,1],\quad(16)$$

To suppress the small correlation values of (16) the Lawley-Hotelling Trace statistic ([27], [28]) $f(\rho_i)=\frac{\rho_i^2}{1-\rho_i^2}$ was proposed in [6]. Our findings support that $f(\rho_i)$ (henceforth be called resemblance values) does suppress smaller values (mostly coming from false alarms), and to further handle the issue of false alarms we employ Benjamini-Hochberg procedure [29], [7] of false discovery rate (FDR) control as follows.

Let our proposed detector impose a threshold τ (to be determined) on resemblance values $f(\rho_i), \forall i=1,2,\ldots,MN$, giving us R as the total number of detections of which W are incorrect (i.e., false alarms). In what follows, $U=\frac{W}{R}$ is the proportion of error committed by our detector. Since we do not know W apriori, U denotes the unobservable random quotient —

$$U=\begin{cases}\frac{W}{R}, \text{ if } R>0,\\ 0, \qquad\text{otherwise.}\end{cases}\quad(17)$$

The FDR, defined by the expectation $\mathbf{E}(U)$, is controlled at a desired level α while maximizing the expectation $\mathbf{E}(R)$. We have $p_1, p_2, \ldots, p_{MN}$ which denote the p-values ($p_i=1-P_{\rho_i}$, where P_{ρ_i} is the cumulative distribution function of resemblance values $f(\rho)$) corresponding to $\{f(\rho_1), f(\rho_2), \ldots, f(\rho_{MN})\}$. FDR control is readily implemented as follows:

- Step 1: define maximum allowable desired FDR bound (on an average) $\alpha \in (0, 1)$.
- Step 2: order the p-values in ascending order yielding $\{p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(MN)}\}$.
- Step 3: let $f(\rho_{(r)})$ be the query window on target corresponding to $p_{(r)}$. Let β be the largest r for which $p_{(r)} \leq \frac{r}{MN}\alpha$.
- Step 4: identify the threshold τ corresponding to $p_{(\beta)}$, and predict that the query windows (centered at $\mathbf{x}_i$) having $f(\rho_i)$ above τ contain instances of the query object Q.

After the significance testing with τ as above we perform non-maxima suppression [30] as the last step to eliminate duplicate detections close to an already identified MCS peak.

Though the algorithm is resilient to minor scale and rotation perturbation of the query, severe changes in its pose requires an altogether different strategy. Here, we handle two kinds of in-plane query distortions — scaling and rotation. In contrast to the setup of [6] we do not scale the target image features. Instead, we scale and rotate query features and leave the target features untransformed (for computational reason). Once we obtain the MCS values, for all scales and orientation, at a particular sliding window location, we select the right scale and orientation by doing maximum likelihood estimation following [6].

B. Fast Target Processing for Rapid Query Search

To mitigate prohibitive computational load $\mathcal{O}(M \times N \times m \times n)$ owing to straightforward sliding window search, Seo et al. mostly relied on evaluating MCS on a sparse grid (coarse-to-fine) search [6], or saliency based pruning techniques [19] to aggressively reduce search space. Though valid and partly effective, such approximate search methods run the risk of missing detection peaks — a fact that often manifests itself as missed detection or as imprecisely located/oriented bounding box on the target image.

Besides pruning based approximate approaches (e.g., active learning in [31]) in the past, exact search of decision function maxima with branch and bound search schemes have also been investigated for object detection. Though branch and bound techniques [32], [33] are designed to converge to global maximum of decision function they are specifically designed for (bag-of-word style) histogram features, and it is not directly evident how to extend such frameworks to MCS based decision rule (the signs of feature elements can not be known a-priori to design sign based integral images [32], [33]).

1) Exact Acceleration of Matrix Cosine Similarity: Our intention remains going beyond the sliding window scheme to get rid of the $m \times n$ factor in the complexity of $\mathcal{O}(M \times N \times m \times n)$, and at the same time sticking to the exact computation of MCS. We proceed by first reshaping query feature $\mathbf{F}_Q \in \mathbb{R}^{d \times (mn)}$ to $m \times n \times d$ feature matrix, and in a similar fashion we reshape $\mathbf{F}_{T_i} \in \mathbb{R}^{d \times (MN)}$ to the size $M \times N \times d$. With slight notational abuse we retain the same nomenclatures for the reshaped query and target features. Paying this polite nod, we write the MCS expression (16) in the following fashion, noting that the numerator is (feature channel wise) cross-correlation between $\mathbf{F}_Q$ and $\mathbf{F}_{T_i}$ which can be efficiently computed in Fourier domain:

$$\rho(\mathbf{F}_Q, \mathbf{F}_{T_i}) = \sum_{c=1}^{d}\sum_{q=1}^{n}\sum_{p=1}^{m} \frac{\mathbf{F}_Q(p,q,c)\mathbf{F}_T(x_{i1}+p, x_{i2}+q, c)}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_T(x_{i1}:x_{i1}+m, x_{i2}:x_{i2}+n, 1:d)\|_F}, \tag{18}$$

$$= \frac{\sum_{c=1}^{d}\sum_{q=1}^{n}\sum_{p=1}^{m} \mathbf{F}_Q(p,q,c)\mathbf{F}_T(x_{i1}+p, x_{i2}+q, c)}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_T(x_{i1}:x_{i1}+m, x_{i2}:x_{i2}+n, 1:d)\|_F}, \tag{19}$$

$$= \frac{\mathrm{IFT}\{\sum_{c=1}^{d} \overline{\mathrm{FT}}\{\mathbf{F}_Q(:,:,c)\}\mathrm{FT}\{\mathbf{F}_T(:,:,c)\}\}}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_T(x_{i1}:x_{i1}+m, x_{i2}:x_{i2}+n, 1:d)\|_F}, \tag{20}$$

where $\mathrm{FT}\{\cdot\}$, $\mathrm{IFT}\{\cdot\}$, and $\overline{\mathrm{FT}}\{\cdot\}$ denote Fourier transform, inverse Fourier transform, and conjugated Fourier transform respectively. Two important facts are worth mentioning at this point. First, correlation can directly be achieved by multiplying one Fourier transform with another, conjugated. Second, since Fourier transform is a linear operator, one can perform the channel wise correlation right in frequency domain followed by channel wise summation [16].

However, two important distinctions with [16] exist in the proposed acceleration. First, we do not compute spatial correlation by converting it into equivalent convolution problem in frequency domain (by 180° rotation of query); correlation between two signals can be directly achieved in frequency domain by first taking Fourier transform of both signals, and then taking complex conjugate of just one of them followed by point by point multiplication (Hadamard product) in frequency domain. Second, MCS has also a normalization factor $\|\mathbf{F}_{T_i}\|$ in the denominator (16) requiring a different strategy for faster computation that Dubout *et al.* did not face in [16]. The trick to efficient computation of the normalization factor lies in a precomputed area sum table of target feature vectors' L_2 norm. Due to the presence of $\|\mathbf{F}_{T_i}\|$ in the denominator of MCS (16) one can not carry out the entire computation in Fourier domain. The target feature elements in $\|\mathbf{F}_T\|$ are individually squared and summed across all channels followed by an integral image construction. Next, one goes through this integral image and compute $\|\mathbf{F}_{T_i}\|^2$ in constant time with three arithmetic operations. This is followed by squaring and dividing the numerator by denominator to yield $f(\rho_i)$.

Indeed, a similar technique has found application in a somewhat dated but absolutely relevant work of J. P. Lewis [34]. However, Lewis used a different form of correlation [1] to consider, and the idea proposed in his work does not involve multichannel features, nor the multiscale and multioriented pattern detection. So, the methodology described in [34] is roughly comparable to a special case of a much more general framework proposed here when the number of feature channels reduces to one, and detection happens at single scale and orientation.

Fig. 6 shows the details of accelerated computation of MCS (at single scale and orientation). The descriptors are computed

[1]The idea of correlation manifests itself in various forms and definitions, and quite rightly there exists at least 13 distinct ways to look at the definition of correlation [35].

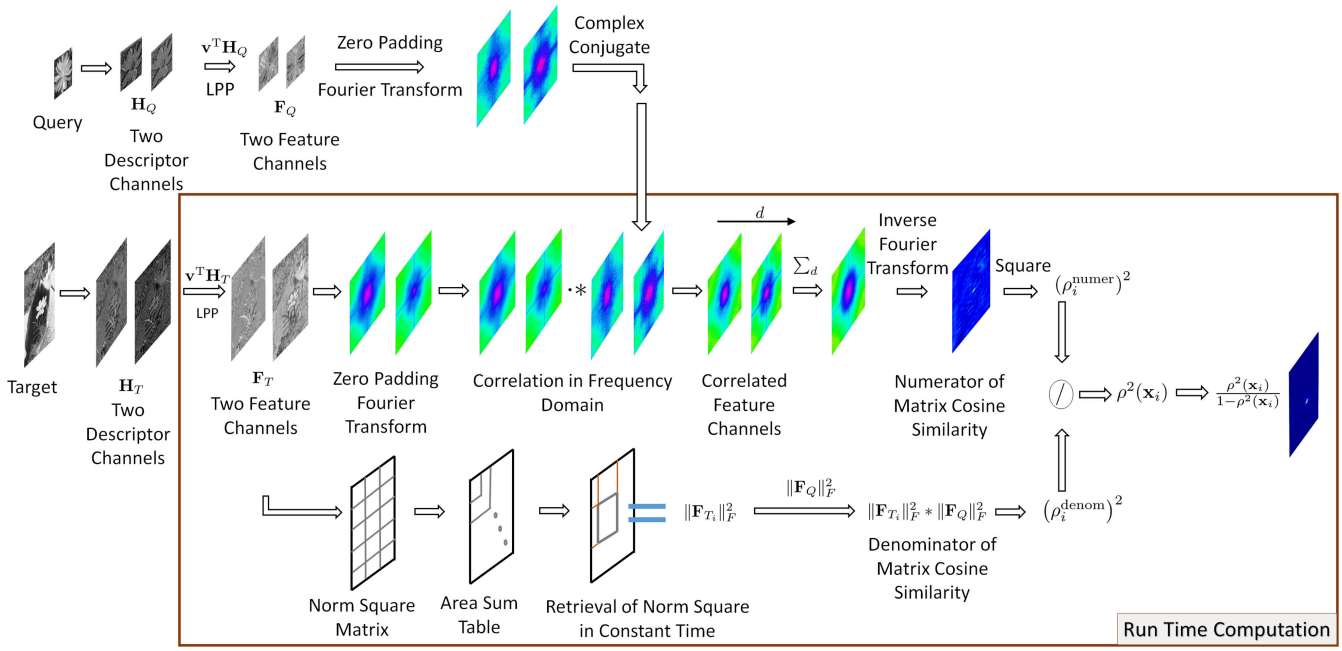


Fig. 6. The illustration of the fast detection algorithm resulting into exact acceleration of matrix cosine similarity computation

a priori, — from both query as well as target — and are treated as parts of the data set following the setup of Lampert [36]. We extract query features, scale and rotate them, do zero padding to bring each feature channel (matrix) to a pre-defined DFT size, apply forward Fourier transform on each feature channel followed by complex conjugation. However, for target T, the transformation on computed features is nil to keep the runtime cost at a bare minimum, only one forward Fourier transform in each feature channel is applied.

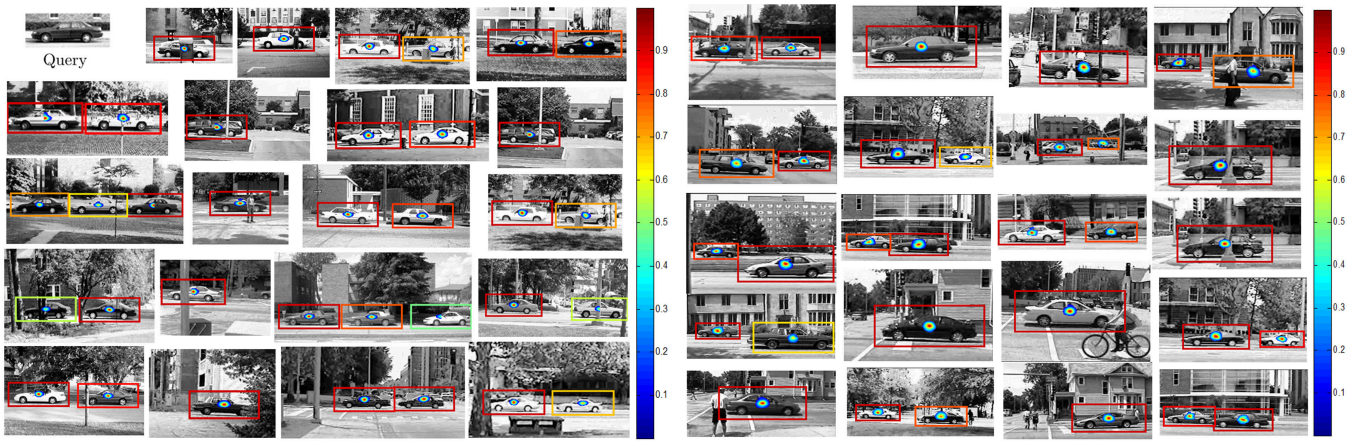
2) Implementation & Computational Time Analysis: To sweep $m \times n \times d$ query window over all locations of $M \times N \times d$ target array one requires to check $(M-m+1)(N-n+1)$ windows for potential objects. A direct evaluation of (16) requires, in each of such sliding windows, first, element by element product followed by summation (in each feature channel) for numerator giving us roughly $2dmn$ computations; and second, similar operations for norm in the denominator produces again (roughly) $2dmn$ computations. Combining the major components and considering total a configurations (equalling to the number of scales times number of orientations), we write the following total computation cost for sliding window scheme.

$$C_{\text{SW}} \approx 4d \cdot a \cdot (M-m+1) \cdot (N-n+1) \cdot m \cdot n. \quad (21)$$

Note here, operations like division and Lawley-Hotelling transformation are of the order of $\mathcal{O}(MN)$, and hence negligible. Before we derive the exact computational cost for proposed fast detection methodology, we note that correlation performed by means of DFT is circular rather than linear, which we require. The difference lies in the fact that circular correlation is an aliased version of its linear counterpart. As long as the DFT matrix is large enough the resulting circular correlation will equal the linear correlation. This is ensured by padding each query feature channel $(m \times n)$ and corresponding target feature channel $(M \times N)$ with sufficient zeros so that zero padded arrays are at least as large as $(M+m-1) \times (N+n-1)$. We assume the zero padded DFT size is $(M_p \times N_p)$, where $M_p \geq M+m-1$, and $N_p \geq N+n-1$. It is also worthwhile to mention that a good practice is in keeping the DFT size at a power of 2 for leveraging the inherent efficiency of Fourier transform. Of course, with variable target size one can go with mixed-radix DFT. Now, a single forward/backward DFT involves computational cost $C_{\text{DFT}} \approx 2.5 M_p N_p \log_2(M_p N_p)$ as in [16]. We need d forward DFT for target plus one inverse DFT for each configuration of the query (Fig. 6). Hence, considering the cost of Hadamard product across all feature channels for all configurations ($C_{\text{prod}} = daM_pN_p$) followed by the cost of channel wise summation ($C_{\text{sum}} = daM_pN_p$), we write the total cost for numerator of (20) as,

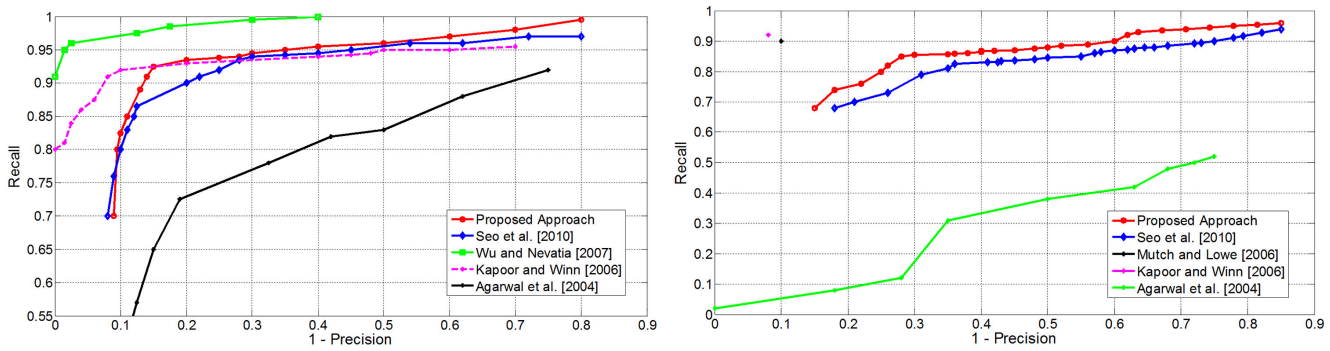
$$\begin{aligned} C_{\text{numer}} &= dC_{\text{DFT}} + C_{\text{prod}} + C_{\text{sum}} + aC_{\text{DFT}}, \\ &\approx (d+a)2.5M_pN_p \log_2(M_pN_p) + 2adM_pN_p. \end{aligned} \quad (22)$$

Producing the norm squared integral image from target feature matrices requires time complexity $2dMN$, because each feature element is squared and summed over all d-channels. Retrieval of $\|\mathbf{F}_{T_i}\|^2$ corresponding to the sliding window location $\mathbf{x}_i$ in target T happens in constant time $\mathcal{O}(1)$ with only three arithmetic operations yielding $3MN$ cost. Next, squaring the numerator followed by the division by the product of $\|\mathbf{F}_{T_i}\|^2$ and the constant term $\|\mathbf{F}_Q\|$ are again three constant time operations per configuration. The construction of $f(\rho)$ involves another two constant operations (subtraction in denominator and division) per configuration. Taking all this information into



(a) (b)

Fig. 7. Example detections on UIUC car test set [37] are shown here. (a) Single scale car detection (the query image is shown top left), and (b) Multiscale car detection (the same query image as used in single scale experiment is used here). The FDR α is set at 1%. The $f(\rho)$ values above the threshold τ corresponding to α is embedded inside the displayed bounding box. A red bounding box indicates highest resemblance to query image, and for other colors the colormap shown right depicts relative resemblance.



(a) (b)

Fig. 8. Precision recall curves obtained from the evaluation of our proposed methodology on UIUC single scale car test set (left), and UIUC multiscale car test set (right) in comparison to other training based state of the arts [38], [39], [37], [40] as well as training-free state of the art methodology [6].

account, we end up with the following cost of denominator computation across all d feature channels,

$$\begin{aligned} C_{\text{denom}} &= 2dMN + 3MN + 3aMN + 2aMN, \\ &= (2d + 3 + 5a)MN. \end{aligned} \tag{23}$$

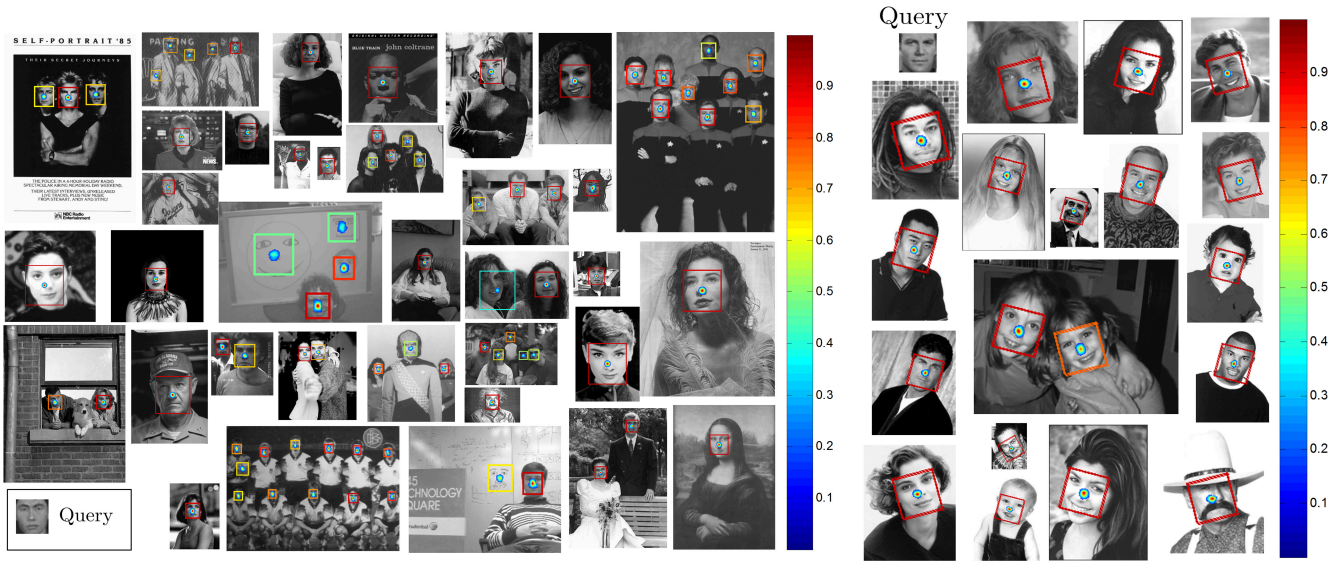
Considering the division of numerator by the denominator the total cost C_{proposed} of MCS computation boils down to —

$$\begin{aligned} C_{\text{proposed}} &= C_{\text{numer}} + C_{\text{denom}}, \\ &\approx 2.5(d + a)M_pN_p \log_2(M_pN_p) \\ &\quad + 2adM_pN_p + (2d + 3 + 5a)MN. \end{aligned} \tag{24}$$

The key observation here is that the proposed detection methodology has made the computational cost independent of the query size $m \times n$ for a *fixed* DFT size $M_p \times N_p$. Large computational mileage results from this fact especially when the query size changes as long as the maximum required DFT size is less than the fixed DFT size. There is further advantage when $d + a << ad$, i.e., with increasing number of query templates, and feature channels, one reaps increasing benefit in comparison to sliding window scheme. Indeed, Dubout *et al.* [16] have achieved almost 13 times theoretical speedup by leveraging Fourier transform in their part based detection process. In our setup, if we plugin typical values for the cost parameters in the final cost expression (24) we get the following result: for $m = 64, n = 64, M = 128, N = 128, a = 1, d = 5$ we get theoretical speedup 20, and for $M = 256, N = 256$ the speedup is 41. Table III gives some ideas of achievable speedup with our (unoptimized) implementation.

V. Experimental Setup, Results & Discussions

In this section, we evaluate the proposed low dimensional features along with runtime performance of the proposed detection methodology. All the experiments are done in a standard desktop machine with 8 GB RAM, Intel Core i7-2600 CPU @3.40 GHz using standard MATLAB functions with no GPU support. Of course, our proposed methodology is general enough to avail of the benefit of GPU computation which would result in even shorter computation time.



(a) (b)

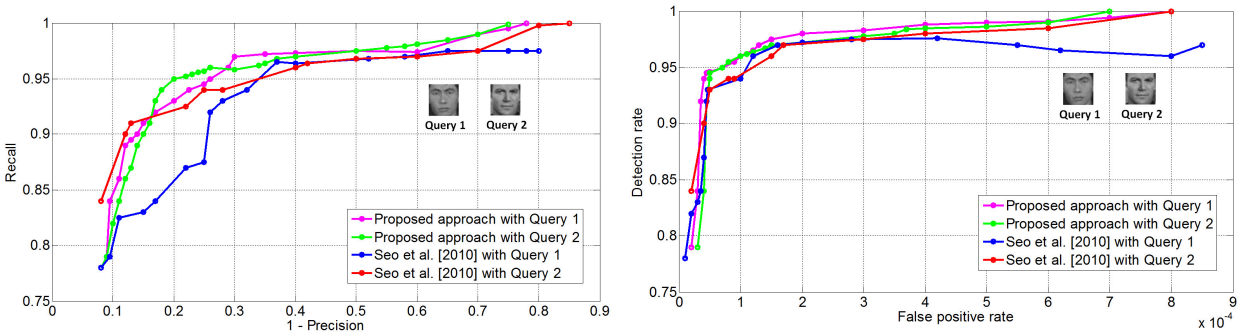
Fig. 9. Face detection in MIT-CMU face data set [41] is illustrated in the figure above. (a) Example detections along with scale estimation are shown using a query face (bottom left). (b) Sample detections along with pose estimation are shown when the scales as well as orientations of the query both vary in target images. In both the experiments, the FDR α is set at 1% to determine the threshold τ. The thresholded $f(\rho)$ is shown inside the bounding box. The correct bounding box results from the maximum likelihood estimate of probable set of scales and orientation. The colormap on right is a mapping between color of bounding box and the measure of resemblance in case of multiple detection; the red means highest resemblance.

In the first part, we have evaluated our methodology on three benchmark data sets: UIUC car data set [37], MIT-CMU face data set [41], Shechtman's general object data set [9], and Caltech 101 data set [42], [43], [44]. The input to the algorithm is a query image with a single dominant object present, e.g., face or car, plus a target image. The typical output of our proposed methodology is a set of bounding boxes drawn around the detected object of interest. By bounding box we mean the smallest possible rectangle drawn around the detected object in the target image. We evaluate the object detection algorithm following the criterion described in [37]: if the detected region overlaps considerably with the ground truth we accept the output of the algorithm as true positive (or correct detection). Otherwise, the detection is regarded as false alarm. With each pair of recall and precision value collected by varying the false discovery rate α, we draw precision-recall and/or receiver operating characteristics (ROC) curves. Also, for the purpose of comparison with other competing detection methodologies we report *detection equal error rate* which is same as recall rate when recall is equal to precision.

A. UIUC Car Data Set

This gray-scale image data set comprises training (500 car and 500 noncar images) and test sets. The test set contains car images at i) same scale (with 170 images of 200 cars, some images having multiple cars of size approximately 100×40 pixels matching closely with the size of the cars in the training sets), and at ii) multiple scales (with 108 images of 139 cars at various sizes where the ratio of scales between largest and smallest car being around 2.5). Since this paper focuses on one shot object detection task, we use a single car from the training set as our query image.

The LARK descriptors at pixel $\mathbf{x}_i$ are computed over a 9×9 patch centered at $\mathbf{x}_i$ yielding 81-dimensional local descriptors $\mathbf{h}_i$. The smoothing parameter σ for computing LARK has set to 1.0; the value of σ in the estimation of pairwise affinity $\mathbf{K}_{ij}$ (15) between $\mathbf{h}_i$ and $\mathbf{h}_j$ also remains the same. Following locality preserving projection we reduce the dimension of LARK descriptors from $l = 81$ to $d = 5$ by choosing the 5 trailing eigenvectors of (4). It is observed that selecting more eigenvectors does not produce noticeable change in the detection performance. Performing a significance test by setting the FDR $\alpha = 1\%$ we obtain the threshold τ for each test example. Fig. 7(a) shows an example of single scale car detection. In case of multiscale detections, as already mentioned in Section 4.2, we do not enforce any feature transform on target features $\mathbf{F}_T$. The query features $\mathbf{F}_Q$ are scaled as much as 2.5 times for robust detection of objects. We use 0.5 times to 2.5 times scaling of query features by a step size of 0.2. The detection performance of multiscale analysis is shown in Fig. 7(b). The performance of our algorithm is reported after aggregating the results of multiple query images. For a particular threshold τ we obtain a set of precision-recall values for the whole query set which we average to obtain a single precision-recall pair, and next, by varying τ we draw the precision-recall curve in Fig. 8(a)-8(b). The overall performance shows improvement as a consequence of preserving locality in derived features. The proposed approach presented here has also been compared with training based approaches in Table I. The results show that our training-free methodology has been able to take the performance of one-shot detection close to some of the training-based ones.



(a) (b)

Fig. 10. Evaluation of proposed detection technique on MIT-CMU face data set in comparison to [6]: (a) precision-recall curve, (b) ROC curve

B. MIT-CMU Face Data Set

This is also a gray-scale image data set and we have evaluated our methodology on the same subset of images as done by Seo *et al.* [6]. The motivation behind using MIT-CMU data set is to subject our algorithm to severe scale changes (up to as much as a scale factor of 5), and large in-plane rotation of the query pattern as well. The test set consists of 43 gray-scale images (list given in [6]) containing a total of 149 frontal faces, occurring at various scales, and 20 gray-scale images having faces at unusually large ($> 60°$) angular orientation. The query faces used for detection as shown in Fig. 9(a) and 9(b) each has a size 61×61. As has been done in [6], we do not resize the target image to bring it at the same scale as that of the query. Instead, we engage in a multiscale and multiorientation search for the query to achieve correct detection optimizing over its pose parameters. Specifically, for a particular scale we search over all angular orientations, from $0°$ to $360°$, with an interval of $30°$. Parameters like smoothing parameters (h), LARK descriptor size (9×9), number of eigenvectors d for dimensionality reduction, and FDR α remain the same as the ones used in UIUC car data set.

Figures 9(a) and 9(b) show the efficacy of our proposed method. We are able to detect faces at various scales with a tight bounding box. The rotated faces are also detected with correct localization and adequate orientation — the displayed results show the correct angle estimated for the oriented face. Since our aim is to detect visually similar instances we have also been able to detect faces drawn on a white board. Following the evaluation scheme we have presented our result in the form of precision-recall and ROC curves in Fig. 10(a)-10(b) with two query faces.

C. General Object Data Set

In this section, we present the performance of our algorithm on color images. In Shechtman and Irani's general object data set [9] we have applied the proposed concepts to match pose *symbols* of humans with relevant human poses in general photographs (Fig. 11). Several challenging query and target pairs are taken from categories like flowers, heart symbols, peace symbols, and faces (Fig. 12). We follow the similar parameter settings like previous experiments except being little cautious with FDR ($\alpha = 0.5\%$) to deal with false positives in a more conservative fashion. To study color information one

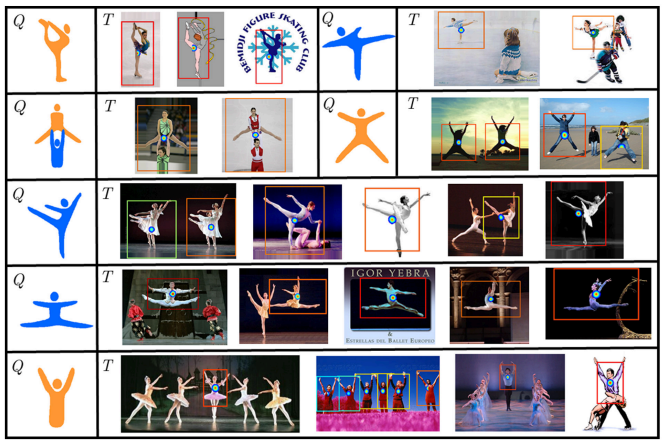


Fig. 11. The human pose symbols as query objects are detected in real life photographs in Shechtman and Irani's general object data set [9]. The query symbols are displayed in the Q panel, and corresponding target detections are shown on right in the T panel. We set the FDR α at 0.5% to deal with false positives conservatively.

can consider several color space models like RGB, YCbCr, and CIE L*a*b*. Our experiments support the findings already reported in [6] and [9], that CIE L*a*b* color model is the most discriminatory. In fact, the luminance channel alone is sufficient to distinguish the object from the clutter in most of the cases as shown in precision-recall and ROC curves in Fig. 13(a)-13(b). In [6], Seo et al., have proposed the use of Canonical Cosine Similarity (CCS) to combine all three color channels for improved detection performance. We endorse their view but at the same time we note that the resulting performance gain as seen from precision-recall and ROC curves is not terribly significant. This comes as no surprise because the structural information (excellently captured by LARK descriptors) alone is enough to compare the visual geometry of query and target, and it is readily available in the luminance channel.

We have compared the performance of present features with other state of the art descriptors like *GLOH* [45], *Shape Context* [46], *SIFT* [18] using the implementation in [45].

TABLE I
DETECTION EQUAL ERROR RATES ON UIUC CARS AND MIT-CMU FACES (MULTISCALE AND MULTI-ORIENTATION)

Datasets	Proposed Approach	Training Based Approaches				
		Agarwal *et al.* [37]	Mutch & Lowe [40]	Kapoor & Winn [39]	Lampert *et al.* [32]	Wu & Nevatia [38]
UIUC Single Scale	90.76	77.08	99.94	94.00	98.5	97.6
UIUC Multiscale	79.01	44.00	90.60	93.50	98.60	-
MIT-CMU Faces	91.24	-	-	-	-	-

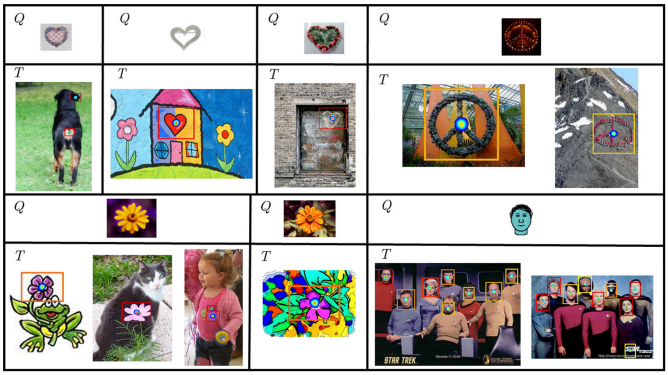


Fig. 12. More examples from Shechtman and Irani's general object data set [9]: Query objects (heart symbol, peace symbol, flower and sketch of human face) are displayed in Q panel; setting the FDR $\alpha = 0.5\%$ we show the corresponding detections in T panels just underneath the relevant Q panels.

We have computed all the local descriptors as densely as possible. To facilitate a fair comparison among the descriptors we have maintained the proposed way of matching in rest of the detection process. In other words, we have carried out the experiment on the data set by replacing the LPP features with these descriptors but keeping the rest of the steps the same. The proposed graph-based dimensionality reduction technique is able to robustly capture the local image structure as clearly visible in the performance curves of Fig. 13(a)-13(b).

The proposed detector achieves a detection equal error rate of 84.4% on this data set. In contrast, self-similarity descriptor when densely computed and used in our matching framework yields a detection rate of 79.2%. Saliency based pruning technique to remove redundant and noisy features followed by nearest neighbor voting based matching [47] improves the detection to 82.7% but that gain comes with considerable computational cost as also observed by Chatfield *et al.* [47]. Besides pruning of features, another factor that further increases the runtime is computation of self-similarity at multiple scales over a Gaussian image pyramid. Note, the performance stated above does not contradict the reported 86% detection rates of self-similarity by Schechtman *et al.* [9] because we did not implement the star-graph based ensemble matching that they used in conjunction with self-similar descriptors. In fact, it is not directly evident how ensemble matching performs in terms of false positive rate as well as computational efficiency when compared with MCS based detection, because [9] did not mention the false positive rate (corresponding to reported detection rate) and computation time. Therefore, our evaluation makes the proposed methodology more practical as Figure 13 provides explicitly the estimates of false alarm versus detection tradeoff. The runtime analysis is discussed in Section V-F.

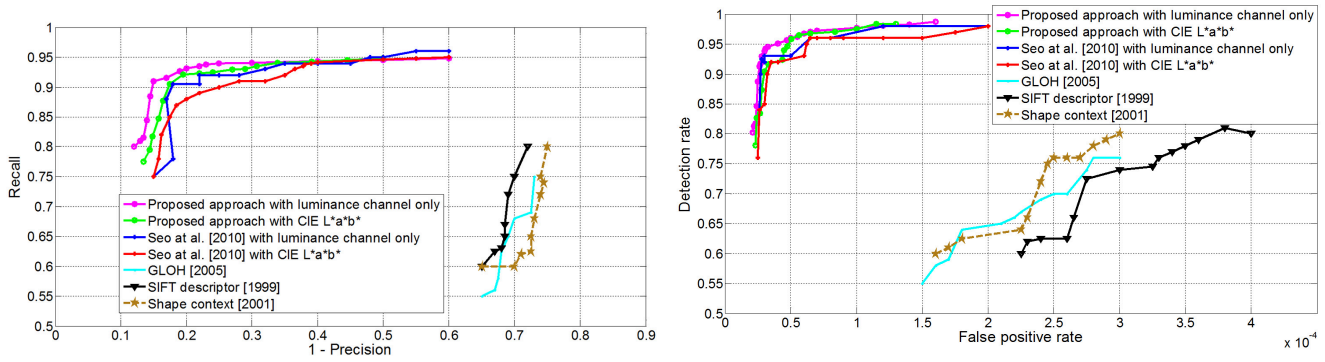
D. Caltech Data Sets

Caltech 101 [42], [43], [44] is a color data set containing 101 object categories. In general, there are 80 to 100 images per category but in some cases number of images may be as high as 800 in a category. Size of each image is roughly 300×200. This data set has a single object present in all the images. An important point to mention is that objects in Caltech data sets vary a lot both in terms of viewpoint (i.e., off-the-plane pose variation) and intra-class pose variation.

In the present work, we study visual similarity between objects with emphasis on efficient detection techniques involving in-plane pose variation (scaling and rotation only). We don't consider class-specific contextual information using training methodologies (to handle intra-class pose variation), nor do we consider huge out of plane viewpoint changes. Caltech 101 has all these variabilities present, however, since it has been widely used in visual recognition task in the past we have presented results of our experiment on this data set (Fig. 14). Past work on Caltech 101 mostly reported performance of training based schemes (with 15 or 30 images as training set) in terms of mean accuracy. Using the proposed embedding technique we obtain a mean recognition rate of 18.5% with a single query (averaged over 15 randomly chosen query images per category). The methodologies proposed by Zhang et al. [48] and Grauman et al. [49] achieved mean accuracies of 21.0% and 18.0% respectively. Here too, we have extracted CIE L*a*b* color model, and used luminance channel alone for subsequent feature computation and detetcion/matching task as explained in case of *General Object Data Set*. Using all LARK channels and leading few feature channels of PCA we obtain mean accuracy figures as 15.7% and 17.2% respectively. Further evaluation of dimensionality reduction is described in the next section.

E. Performance Analysis of Embedding Techniques

Considering our proposed contribution it becomes imperative to study the performance benefit of the locality preserving embedding in contrast to raw LARK channels as well as other embedding technique. For that purpose we have introduced Table II that shows results produced by different components related to the present detector. The proposed LPP features derived from LARK, shown in fourth column of Table II,



(a) (b)

Fig. 13. Evaluation of proposed detection technique on Shechtman-Irani general object data set [9]: on left, precision-recall curves are shown, and on right the ROC curves show the performance of the proposed algorithm along with [6], SIFT [18], GLOH [45], and Shape Context [46]. Experiment is conducted using only luminance channel as well as all CIE L*a*b* channels. In case of CIE L*a*b* channels, canonical cosine similarity [6] is used to fuse information from three channels.

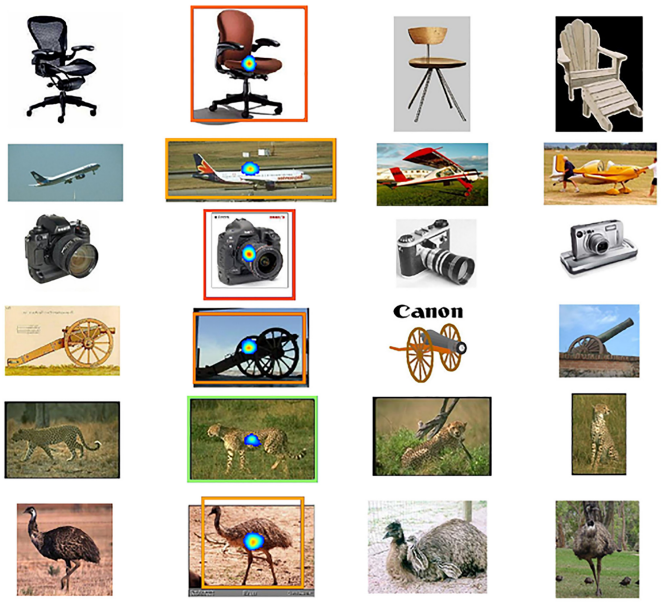


Fig. 14. First column and second column show visually similar matches. Caltech 101 has images with considerable viewpoint and intra-class variation which result into failure in matching shown in third and fourth column. The presence of articulated objects (last two rows) also makes matching difficult in this data set.

works superior to PCA projections of LARK [6] as mentioned in third column.

It is also noted in the second column of Table II that using raw LARK feature channels (with the implementation of [6]) affects the detection performance as also observed by [6]. This is because too much channel information has a derogatory influence on the detector. In contrast, the locality preserving projection aggregates channel information from all LARK channels in such a way that the object contours become very prominent in lower subspace, and the non-contour parts tend to get smoothed out (Fig. 3(d)-(e)). This makes sense because $\mathbf{K}_{ij}$ in locality preserving cost function (1) is built with local aggregation of gradient vectors (Fig. 5(b)), and the subsequent

TABLE II
DETECTION RATES OF RAW LARK AND PROJECTED FEATURES

Data Sets	LARK All Channels [6]	LARK + PCA [6]	Proposed Approach
UIUC Car (Single Scale)	83.92	87.13	90.76
UIUC Car (Multiscale)	73.33	75.47	79.01
MIT-CMU Faces	84.76	86.58	91.24
General Object Data [9]	81.58	83.35	84.41

dimensionality reduction causes strong contours in dominant projections.

Also worth mentioning is the computational benefit that comes with maintaining a low number of feature channels as a result of the embedding, and in the subsequent sections we will see how a few discriminatory feature channels (typically four or five) aids rapid processing of an image in real time.

F. Accelerated Visual Search: Runtime Evaluation & Query Detection in Video

In Section IV, we have derived a theoretical estimate of the runtime of our accelerated search technique. It is worthwhile to mention that for a fixed DFT size we achieve a runtime performance that is independent of the query size. Surely, setting the DFT size too high (by zero padding) to accommodate both query and target inside, may led to somewhat inefficient memory usage. There are various techniques to get around this difficulty. One can work with query and target of reduced sizes, or the more technically correct solution is to perform overlap-add and overlap-save methodology following a mixed-radix implementation.

In our experiments we have presented results comparing the proposed fast object detection with sliding window based scheme [6]. The number of feature channels for evaluating MCS has been kept constant at five. Table III summarizes the runtime in seconds for single scale object detection using two queries of sizes 64×64 and 128×128. The *Power-of-2* implementation assumes the smallest (power-of-2) DFT

Fig. 15. User defined object detection in movie *Charade (1963)*: in leftmost column the user defined query object is highlighted, and example detections are displayed on right. Correct detection has been achieved with high resemblance value even in case of partial occlusion.

size as (M_p, N_p), large enough to hold query plus target sizes, i.e., $M_p \geq M + m - 1$, and $N_p \geq N + n - 1$. Consequently, a 64×64 query and 128×128 target should have a minimum (power-of-2) DFT size of 256×256, and a target 512×512 (or 768×768) should have a DFT size of 1024×1024. We also present runtime in seconds with mixed-radix implementation as part of of our results. Clearly, the results show considerable performance gain rendering the real time object detection feasible. For multiscale search in Table IV, we have categorically used 10 scales, transforming the query features by 0.5 to 2.0 times the original query size (and compared with [6] who transform the target features). For joint multiscale and multiangle detection in Table IV, besides using 10 scales, we have checked 12 orientations (per scale) with equal angular spacing. Table IV reports runtime based on mixed radix implementation of discrete Fourier transform.

Inspired by the famous project *Video Google* by Sivic and Zisserman [50], and later studied by Lampert [36], we have extended our work to user defined query detection in movie videos. Before we go into the experimental details, we point out some novel features of our approach in comparison with previous approaches to *Video Google*. First, our methodology does not require the overhead of bulk codebook creation. The user is free from the tedious task of feature quantization for building a visually descriptive dictionary. Secondly, the proposed detector robustly deals with in-plane variation (i.e., change in scale and orientation), handling extreme clutter, low resolution as well as partial occlusion. We have carried out our experiment on three movie data sets, namely, *Charade (1963)*, *Dressed to Kill (1980)*, and *Ferris Bueller's Day Off (1986)*. The first two movies come with gray scale frames and the last one with color frames. We have processed the following frame sizes for the above movies: 312×240, 320×240, and 416×170. The number of feature channels used is five in number, and we have used FDR $\alpha = 1\%$ to achieve the detection results as shown in Fig. 15, 16, and 17. It is reasonable here to search for the query at 5 scales, 0.8, 0.9, 1.0, 1.1, and 1.2 times the size of the query image selected by the user. We do not consider multioriented detection in this experiment. With the present set of queries we have achieved following detection rates for the three movies: 97.21%, 92.05%, and 88.66%, respectively, as opposed to 93.17%, 84.88%, and 82.25% by [6]. The missed detections result when the query suffers severe off-the-plane distortions resulting in major viewpoint alteration. The proposed method is able to detect queries amidst major in-plane distortions, like significant scale change, partial occlusion, out-of-focus blur, and low resolution. The average time consumed per frame for all the three movies are as follows: 0.131 sec, 0.150 sec, and 0.122 sec, as opposed to 9.581 sec, 11.622 sec, and 10.210 sec by [6]. It is true that codebook-based approaches [50], [33] consume much shorter runtime to process movie frames but two important distinctions exist here. First, our work is training-free and we don't require the user to build a codebook. Secondly, the codebook based approaches [36] do not process the movie frames in linear fashion. In contrast, we are interested in

Fig. 16. Query object detection in movie *Dressed to Kill (1980)*: in top row, leftmost column, the user selects the *bow-tie* as query object, and sample detections are shown in right panels. In second row, leftmost column, the selected *biscuit jar* as query gets detected in subsequent frames in the middle of heavy clutter, scale change, and partial occlusion.

TABLE III
RUNTIME OF PROPOSED FAST OBJECT DETECTION IN COMPARISON WITH SLIDING WINDOW SCHEME

Query Size (pixels)		64×64				128×128		
Target Size (pixels)		128×128	256×256	512×512	768×768	256×256	512×512	768×768
Sliding Window [6] (sec.)		0.3665	2.8815	15.6680	38.9581	5.4294	45.1551	132.1556
Proposed (in sec.)	Power-of-2	0.0304	0.0886	0.3296	0.3313	0.0929	0.3259	0.3341
	Mixed-radix	0.0184	0.0366	0.1360	0.2429	0.0540	0.1650	0.2492

exact search, and hence, the present methodology processes the frames successively in linear sequence fashion; our task is motivated by the long term goal of real time object detection with smart phone cameras or mobile devices when the frames may not be available a-priori.

G. Discussions, Current Trends and Future Directions

The primary difference of LPP features from other features such as SIFT [18], GIST [51], or HOG [25] is the fact that proposed features do not have any geometric invariance like rotational or scale invariance. We encode the image geometry robustly without considering invariance and transfer the rotation/scale considerations to matching/detection phase. While matching a query image with all the parts in a bigger target image, the built-in invariance in the descriptors – though capable of handling off-the-plane pose variation – often leads to too many false positives. Indeed, Seo et al. [6] have observed that for one shot detection task LARK descriptors sacrifice such invariance in exchange of superior localization performance when compared with HOG, GLoH [46], SIFT. One disadvantage in using the current framework is that our proposed detector can handle minor out-of-plane transformation, but severe such viewpoint variation if present can go undetected (see Fig. 18). In such cases the best strategy is to use keypoint based detector with RANSAC matching technique but such methodologies being prone to false alarm often require associated geometric constraints verification.

LARK descriptor in spirit is somewhat close to Local Binary Pattern (LBP) [52], as LBP also captures neighborhood information converted to binary figures by appropriate thresholding. Instead, LARK captures more sophisticated attribute of the signal by measuring the geodesic properties of the neighborhood with respect to the center pixel. Also, LBP works directly on the intensities whereas LARK collects gradient vectors to estimate the locally dominant orientation. Since it works on gradients, LARK is robust to photometric or brightness variation in the image. How other descriptors perform in visual recognition if projected in similar fashion

TABLE IV
RUNTIME OF FAST OBJECT DETECTION WITH POSE ESTIMATION IN COMPARISON WITH SLIDING WINDOW SCHEME

Pose estimation for Query size 64×64 pixels		Different Target Sizes (in pixels)			
		128×128	256×256	512×512	768×768
Multiscale Search Time	Sliding Window [6] (in sec.)	4.607	34.341	182.475	448.740
	Proposed (in sec.)	0.116	0.380	1.718	2.248
Multiscale, Multiangle Search Time	Sliding Window [6] (in sec.)	53.255	398.788	2140.610	5145.767
	Proposed (in sec.)	1.317	4.068	20.142	24.756

Fig. 17. Detection results in movie *Ferris Bueller's Day Off (1986)*: in top row, leftmost column, the user selects the *wall painting* (within camera focus), and subsequent detections include cases with heavy out-of-focus instance and partial occlusions. In second row, the selected *jersey number* is detected against considerable geometric distortion. Lastly, in the third and fourth rows, we see the *red-wing logo* detected in a perfect manner on the T-Shirt despite some challenging distortions like scale, and even aspect ratio.

remains an interesting open question. Furthermore, robust estimation of dominant orientation along with consideration of local geometry during dimensionality reduction make the feature selection robust to the presence of noise – a fact that encourages natural extension of this work to feature extraction from noisy and low-resolution images.

A recent research trend is to obtain mid-level features from low-level descriptors which seem to work well for bag-of-word model. For example, VLAD and Fisher vector summarize the local descriptors over a relatively bigger neighborhood. In principle, they assign the local descriptors to the code elements of a visual codebook, derived by either K-Means clustering (VLAD) [53] or Gaussian mixture model (Fisher vector) [54]. So far, such approaches have been limited to histogram features like rotationally invariant (HOG) or scale invariant (SIFT), what happens if quantization is done with LARK or LARK+LPP features remains an interesting research direction to explore. Assuming an object constitutes of various parts and subparts has led to considerable increase in detection performance, and the notable contribution in this area has been made by Felzenszwalb et al. [55].

With the success of convolutional neural network and deep learning in image classification a new era has started in learning features. Feature learning itself is connected to a few questions of great philosophical interest: how do we learn *representations* [56] that are key to effective perception, like depth, color, shape, texture, light? How such representations can be adapted across domains? In fact, the structured nature of objects are recently modeled with commendable success by deep architectures [57]. With deep learning one can go beyond specifically designed features and learn them with efficient algorithms in unsupervised or semi-supervised fashion, as well as in hierarchy (see [58], [59] for general introduction). In this context, a paper of direct interest in relation to the present work would be [60] by Hinton et al. where dimensionality reduction has been achieved with multilayer neural network. However, deep learning at present requires very sophisticated hardware configurations and equally complicated software setup with a long training time. Also, the availability of a large training set seems to be a crucial factor for it to succeed though research in this area is still in progress.

With ubiquitous presence of mobile devices a new research area [61], [62] is fast emerging where one would like to search objects on mobile devices. Future work involves making the feature computation very fast especially on low-powered, memory constrained devices like mobile phones. In this work we have used the graph Laplacian for embedding the high dimensional LARK descriptors into low dimensional manifold,

Fig. 18. Limitations are shown with example *table* images from Sun data set [63]. A query table is shown top left. The proposed methodology is limited by its inability to detect instances which have suffered severe out-of-plane viewpoint alterations (second and third from left).Since our detector seeks visual similarity for matching, it is not able to detect tables with intra-class variations (last two from right).

but we also note that the graph Laplacian has found application in segmentation of user defined object from the background [22]. Hence, it is reasonable to explore the use of our definition of the graph Laplacian in the joint task of detection and segmentation.

VI. Conclusion

In this paper we have studied visual similarity between two images which could lead to robust and efficient object detection. Given a single query object, searching the same in a bigger image is a hard task given various pose and scale variations that the object undergoes. We have addressed such concerns in this work. Typically, the descriptors traditionally used in visual recognition for encoding image geometry have important information in all the channels. Extracting a useful gist of them without sacrificing the descriptor's discriminative power is not straight forward. To address such concern, we have studied a graph based dimensionality reduction method by combining local signal patterns with global context, preserving discriminative details of image patterns for one shot object detection and concurrent pose estimation. The algorithm described is quite general; one can integrate the methodology with any descriptors depending on the application in hand. The results with LARK descriptors show LPP improves detection in comparison to PCA by being aware of local structure, thereby making correct estimate of the object location, its scale, and orientation. Since the sliding window based detection scheme is very slow in practice, we have proposed a faster method to evaluate the decision rule. In contrast to approximated visual search (e.g., in pruning based methods), the proposed fast detection technique uses frequency domain to perform correlation computation along with area sum table to arrive at the exact acceleration of MCS computation.

References

[1] K. Grauman and B. Leibe, "Visual object recognition," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 5, no. 2, pp. 1–181, 2011.

[2] Imagenet: Large scale visual recognition challenge 2014. [Online]. Available: http://image-net.org/challenges/LSVRC/2014/index

[3] N. Vasconcelos and M. Vasconcelos, "Minimum probability of error image retrieval: From visual features to image semantics," *Foundations and Trends in Signal Processing*, vol. 5, no. 4, pp. 265–389, 2012.

[4] (2013) Fine-grained classification challenge. [Online]. Available: https://sites.google.com/site/fgcomp2013/

[5] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *arXiv preprint arXiv:1306.5151*, 2013.

[6] H. J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1688–1704, 2010.

[7] H. J. Seo and P. Milanfar, "Action recognition from one example," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 867–882, 2011.

[8] S. K. Biswas and P. Milanfar, "Laplacian object: One-shot object detection by locality preserving projection," in *Proc. IEEE Conf. ICIP*, 2014.

[9] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *Proc. IEEE Conf. CVPR*, 2007, pp. 1–8.

[10] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *Proc. IEEE Conf. CVPR*, 2008, pp. 1–8.

[11] K. Chatfield, J. Philbin, and A. Zisserman, "Efficient retrieval of deformable shape classes using local self-similarities," in *Proceedings of IEEE ICCV Workshops*, 2009, pp. 264–271.

[12] A. Vedaldi and A. Zisserman, "Self-similar sketch," in *Proc. Conf. ECCV*. Springer, 2012, pp. 87–100.

[13] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.

[14] T. Deselaers and V. Ferrari, "Global and efficient self-similarity for object classification and detection," in *Proc. IEEE Conf. CVPR*, 2010, pp. 1633–1640.

[15] J. J. Hull, "A database for handwritten text recognition research," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 16, no. 5, pp. 550–554, 1994.

[16] C. Dubout and F. Fleuret, "Exact acceleration of linear object detectors," in *Proc. Conf. ECCV*, 2012, pp. 301–311.

[17] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems*, 2004, pp. 153–160.

[18] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[19] H. J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *Journal of vision*, vol. 9, no. 12, p. 15, 2009.

[20] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[21] M. W. Mahoney, L. Orecchia, and N. K. Vishnoi, "A local spectral method for graphs: With applications to improving graph partitions and exploring data graphs locally," *Journal of Machine Learning Research*, vol. 13, no. 1, p. 2339, 2012.

[22] S. Maji, N. K. Vishnoi, and J. Malik, "Biased normalized cuts," in *Proc. IEEE Conf. CVPR*, 2011, pp. 2057–2064.

[23] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering." in *NIPS*, vol. 14, 2001, pp. 585–591.

[24] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. CVPR*, 2005, pp. 886–893.

[26] H. J. Seo and P. Milanfar, "Face verification using the lark representation," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 4, pp. 1275–1286, 2011.

[27] M. M. Tatsuoka, *Multivariate analysis,*. Macmillan, 1988.

[28] T. Caliński, M. Krzyśko, and W. Wołyński, "A comparison of some tests for determining the number of nonzero canonical correlations," *Communications in Statistics, Simulation and Computation®*, vol. 35, no. 3, pp. 727–749, 2006.

[29] Y. Benjamini and Y. Hochberg, "Controling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B*, vol. 57, no. 1, pp. 289–300, 1995.

[30] F. Devernay, "A non-maxima suppression method for edge detection with sub-pixel accuracy," *Technical report, INRIA*, no. RR-2724, 1995.

[31] R. Sznitman and B. Jedynak, "Active testing for face detection and localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1914–1920, 07 2010.

[32] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: Object localization by efficient subwindow search," in *Proc. IEEE Conf. CVPR*, 2008, pp. 1–8.

[33] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Efficient subwindow search: A branch and bound framework for object localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2129–2142, 2009.

[34] J. Lewis, "Fast normalized cross-correlation," in *Vision Interface*, vol. 10, no. 1, 1995, pp. 120–123.

[35] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[36] C. H. Lampert, "Detecting objects in large image collections and videos by efficient subimage retrieval," in *Proc. IEEE Conf. ICCV*, 2009, pp. 987–994.

[37] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1475–1490, 2004.

[38] B. Wu and R. Nevatia, "Simultaneous object detection and segmentation by boosting local shape feature based classifier," in *Proc. IEEE Conf. CVPR*, 2007, pp. 1–8.

[39] A. Kapoor and J. Winn, "Located hidden random fields: Learning discriminative parts for object detection," in *Computer Vision–ECCV 2006*. Springer, 2006, pp. 302–315.

[40] J. Mutch and D. G. Lowe, "Multiclass object recognition with sparse, localized features," in *Proc. IEEE Conf. CVPR*, vol. 1, 2006, pp. 11–18.

[41] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998.

[42] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *IEEE CVPR Workshop of Generative Model Based Vision (WGMBV)*, 2004.

[43] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[44] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 4, pp. 594–611, 2006.

[45] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[46] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[47] K. Chatfield, J. Philbin, and A. Zisserman, "Efficient retrieval of deformable shape classes using local self-similarities," in *Workshop on Non-rigid Shape Analysis and Deformable Image Alignment, ICCV*, 2009.

[48] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2. IEEE, 2006, pp. 2126–2136.

[49] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Internationl Conference on Computer Vision (ICCV)*, vol. 2. IEEE, 2005, pp. 1458–1465.

[50] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proceedings of the International Conference on Computer Vision*, vol. 2, Oct. 2003, pp. 1470–1477.

[51] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[52] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[53] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 3304–3311.

[54] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 143–156.

[55] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[56] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[57] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," *CoRR*, vol. abs/1310.1531, 2013. [Online]. Available: http://arxiv.org/abs/1310.1531

[58] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009, also published as a book. Now Publishers, 2009.

[59] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014, pp. 580–587.

[60] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504 – 507, 2006.

[61] V. Chandrasekhar, G. Takacs, D. Chen, S. S. Tsai, J. Singh, and B. Girod, "Transform coding of image feature descriptors," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2009, pp. 725 710–725 710.

[62] D. M. Chen, S. S. Tsai, V. Chandrasekhar, G. Takacs, J. Singh, and B. Girod, "Tree histogram coding for mobile image matching," in *Proceedings of IEEE Data Compression Conference (DCC)*, 2009, pp. 143–152.

[63] J. Xiao, H. Hays, K. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3485–3492, 2010.