Linear Support Tensor Machine with LSK Channels: Pedestrian Detection in Thermal Infrared Images

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

***Abstract*—Pedestrian detection in thermal infrared images poses unique challenges because of the low resolution and noisy nature of the image. Here we propose a mid-level attribute in the form of the multidimensional template, or tensor, using Local Steering Kernel (LSK) as low-level descriptors for detecting pedestrians in far infrared images. LSK is specifically designed to deal with intrinsic image noise and pixel level uncertainty by capturing local image geometry succinctly instead of collecting local orientation statistics (e.g., histograms in HOG). In order to learn the LSK tensor we introduce a new image similarity kernel following the popular maximum margin framework of support vector machines facilitating a relatively short and simple training phase for building a rigid pedestrian detector. Tensor representation has several advantages, and indeed, LSK templates allow exact acceleration of the sluggish but de facto sliding window based detection methodology with multichannel discrete Fourier transform, facilitating very fast and efficient pedestrian localization. The experimental studies on publicly available thermal infrared images justify our proposals and model assumptions. In addition, the proposed work also involves the release of our in-house annotations of pedestrians in more than 17000 frames of OSU Color Thermal database for the purpose of sharing with the research community.**

I. Introduction

The computer vision community has made good progress in people and pedestrian detection in natural images and videos in the last ten years [1]–[5]. However, such endeavors in locating pedestrians have mostly been restricted to photographs captured with visible range sensors. Infrared and thermal imaging sensors, which provide excellent visible cues in unconventional settings (e.g., night time visibility), have historically found their use limited to military, security and medical applications. However, with increasing image quality and decreasing price and size, some of the thermal sensing devices are finding commercial deployment for home and office monitoring as well as automotive applications [6]–[9]. Research effort has so far been limited in this domain for building reliable and efficient computer vision systems for infrared thermal image sensors. The objective of this paper is to address this concern.

Thermal image sensors typically have a spectral sensitivity ranging from 7 micron to 14 micron band of wavelength. The capacity of these imaging devices to appropriately capture images of objects depends on their emissivity and reflectivity in a nontrivial fashion. The material and surface properties of the objects control emissivity whereas amount of background radiation reflected by the objects influence their reflectivity [10]. The involvement of multiple factors in the image formation process often leads to various distortions in thermal images, notably, *halo effect, hotspot areas, radiometric distortions* to name a few [11]. Fig. 1 illustrates halo effect in a natural-thermal image pair. Also noticeable is the fact that textures visible on objects often get suppressed in thermal images. This fact has important bearing as far as visual recognition is concerned because the negative examples corresponding to the background tend to be far less descriptive. Fig. 1 also illustrates the challenge involved in detecting foreground objects because of inherently noisy nature of the infrared images. From the representative images it is fair to conclude that a successful visual recognition system must include a strong measure of visual similarity that can overcome the effect of weak and ambiguous image signal as well as a robust noise handling component in the feature computation process.

In this paper we focus our attention on detecting pedestrians, particularly walking at a distance from the camera. Since our primary motivation for studying infrared images comes from automotive applications, we focus our attention and study on building pedestrian detectors without considering tracking information and background model. Of course, the proposed methodology is general enough to include such information toward building more sophisticated models. Ensemble based techniques like boosting and random forest [12], convolutional neural network (CNN) and deformable part model [2] are three widely used approaches toward building an effective object detector. Though extremely fast in runtime, training with boosting and CNN often takes too long to converge, sometimes spanning days. The ready availability of a substantially large clean annotation set is often recommended for feature learning particularly with deep architectures. In this paper, we shall not delve into feature learning (as done in CNN [13]) but focus on a fast and efficient mid-level attribute that can allow clean, simple training phase with reasonably good detection performance. The widely successful pedestrian, and in general, pose detector deformable part model is built upon the fundamental notion of representing templates with the Histogram of Oriented Gradients (HOG) as mid-level representation. Besides HOG, the use of Local Binary Pattern (LBP) (followed by multiscale feature computation using scale approximation, and faster detection with cascades of AdaBoost classifiers [12]) in effective detection of the pedestrians in thermal images is also explored in literature [8]. However, in case of a part based detector the runtime cost associated with the *part* complexities appears to be a major issue. Also, the small size of the pedestrians when they appear far from from the camera does not leave room for the explicit modeling of the parts/limbs. Interestingly, a recent study [14] has shown that with careful design a seemingly naive rigid detector can perform exceptionally well in comparison to its

(a) (b) (c) (d)

Fig. 1. Infrared images are different: natural color images exhibit textures which are suppressed in infrared images (left pair images [15] [1]). As a consequence, many background texture features like trees and buildings may remain relatively nondescriptive (third from left) which complicates the separation of the background in feature space during the learning process. In addition, the high noise adds to the complexity of detecting foreground objects (far right).

advanced counterparts with higher complexities. Our work draws inspiration from their study, and in this paper we revisit the simple but very effective detector of Dalal and Triggs [1], with the following contributions.

Maximum margin matrix cosine similarity with LSK tensors: dealing with heavy noise and artifacts in image signal while performing visual recognition has garnered relatively low attention from the community. This is particularly relevant in infrared domain where sensor noise is high, and feature variability is much less compared to natural photographs. Our objective is to capture local image structure in a stable and reliable fashion. For accomplishing that purpose we advocate the use of Local Steering Kernel (LSK) [16]–[18] as low level image region descriptor. LSK had its genesis primarily in the image denoising and filtering tasks [16], and is also known as Locally Adaptive Regression Kernel, or LARK, following the fact that LARK filter coefficients are computed adaptively following a local regression on neighboring pixel intensities. Using LSK [2] as low level descriptor we propose a tensor representation of our mid-level attribute — the detector template. However in doing so, the geometric invariance (as in HOG [1], [2]) and scale invariance (as in SIFT [19]) are relaxed at the expense of robustness of the descriptor. HOG and LSK both capture local orientation information. However, HOG computes orientation *statistics* over a set of angular directions in a small spatial neighborhood, whereas LSK captures dominant local orientation and is thus more stable in dealing with image noise.

Our contribution is a maximum margin learning methodology that respects and leverages the tensor form of our mid-level representation. Past work has highlighted the effective use of Matrix Cosine Similarity (MCS) as a robust measure for computing image similarity in a training-free, one shot detection scenarios. [17], [20]. Motivated by such findigs we have extended MCS to introduce a maximum margin training formulation for learning a decision boundary that can separate pedestrian from the background. The standard technique followed in the test time for object search, i.e., sliding window based object detection, incurs prohibitive computational cost. To resolve this issue Lampert *et al.* have proposed a branch and bound technique [21], [22]. In our work, we propose a relatively simple but efficient technique to improve the detection time, performing multiscale pedestrian detection in less than a second. The search for pedestrian in a test image proceeds in the frequency domain (using Fourier transform) with integral image based normalization, yielding an elegant framework for extremely efficient and fast pedestrian detection.

Analysis and Annotations: we have demonstrated the efficacy of the proposed methodology on three standard benchmark datasets [23]–[25]. In particular, we have annotated OSU Color-Thermal pedestrian dataset as we could not find a good annotation in the public domain suitable for evaluating various object detection algorithms. To help push the state of the art in this area our work also includes the ground truth annotations of pedestrians in 17088 frames in this dataset[3].

The related work till date explored LSK in various detection scenarios on natural images but stayed limited on two counts. All such past studies i) did not explore or leverage inherent tensor connection of LSK, and ii) did not extend MCS toward a more general, learning based scheme. For example, Seo et al. [17], [26] and Biswas et al. [20] restricted LSK and MCS to the study of training-free, generic one shot object detection. Subsequent investigations by Zoidi et al. [27] reported performance of LSK to detect humans in videos. In a further extension, You et al. [28] used LSK features for learning local metric for ensemble based object detection. Though laudable, in all of such research endeavors, the generalization principle of MCS had been missing. We believe it is promising to show that MCS could be effectively and efficiently integrated with LSK for building large scale learning systems. The whole premise behind this work is that tensor representations when combined with MCS kernel would invariably lead to a reasonably simple and fast training scheme, with rapid and precise localization result.

We proceed with the system overview in the next section.

II. System Overview

Unlike color images that contain multiple color channels infrared images typically have a single channel that we denote by $M \times N$ image matrix $\mathbf{I}$, defined on a rectangular

[1]The left pair of images [15] (available online, 5th August, 2016) is downloaded from: http://www.dgp.toronto.edu/~nmorris/IR/ for academic use

[2]We shall follow the name LSK since it intuitively indicates characteristics of the features

[3]Available for download from the first author's website

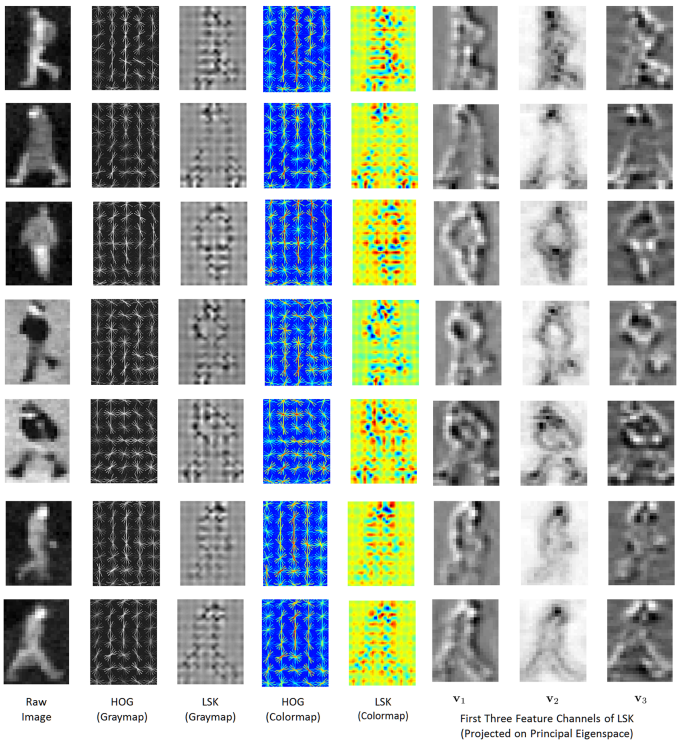


Fig. 2. **LSK Visualization** First column displays raw infrared images of pedestrians. HOG and LSK features are displayed in grayscale (second and third column respectively) as well as in in colormap (fourth and fifth column respectively). LSK is displayed thus after computing them in non overlapping blocks. Columns sixth, seventh and eighth show LSK features after projecting LSK descriptors on three leading principal components.

grid $\Omega \subset \mathbb{R}^2$. We densely compute low-level, l-dimensional descriptors $\mathbf{h}_i \in \mathbb{R}^l$, at each pixel location $\mathbf{x}_i \in \Omega$. Aggregating all '$\mathbf{h}_i$'s together, we form our third-order descriptor tensor $\mathbf{H} \in \mathbb{R}^{M \times N \times l}$ corresponding to image $\mathbf{I}$. The order, or dimension, three in $\mathbf{H}$ alludes to the l channels of the computed descriptor.

Dense computation makes the descriptor highly descriptive no doubt, but at the same time it invites the undue effect of redundancy. To distill the redundancy we reduce the number of channels in $\mathbf{H}$ from l to d by employing principal component analysis. The result is the decorrelated feature tensor $\mathbf{F} \in \mathbb{R}^{M \times N \times d}$, where $d << l$.

From $M \times N \times d$ tensor $\mathbf{F}$ we crop a smaller third order tensor window $m \times n \times d$ corresponding to the ground truth annotation that represents a 'pedestrian'. We follow the similar process for collecting the negative examples that correspond to the 'background'. Specifically, the i-th training example $\mathbf{F}_i \in \mathbb{R}^{m \times n \times d}$ associated with the class label y_i, is first normalized and then used as an input to a maximum margin classification using our proposed kernel function. The objective is to learn a decision boundary to separate pedestrians from background in the tensor feature space. We describe our full methodology starting with the feature computation in the next section.

III. Local Structure Estimation with Steering Kernel

We represent each pixel by a two dimensional coordinate vector $\mathbf{x}_i = [x_{i1}\ x_{i2}]' \in \Omega$. We define the image $\mathbf{I}$ as a function such that $\mathbf{I} : \Omega \rightarrow \mathbb{R}$. The value of image $\mathbf{I}$ at a particular pixel location $\mathbf{x}_i \in \Omega$ is given by the pixel intensity $\mathbf{I}(\mathbf{x}_i)$.

LSK derives its name as well as much of its descriptive power from a steering matrix [16] (also known as gradient covariance matrix or structure tensor) that lies at its heart, and defined at the pixel $\mathbf{x}_i$ as follows:

$$\mathbf{C}_{\Omega_i} = \sum_{\mathbf{x}_i \in \Omega_i} \begin{bmatrix} \frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial x_{i1}}^2 & \frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial x_{i1}} \cdot \frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial x_{i2}} \\ \frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial x_{i1}} \cdot \frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial x_{i2}} & \frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial x_{i2}}^2 \end{bmatrix}, \quad (1)$$

where Ω_i is the rectangular window centered at $\mathbf{x}_i$. In theory, the steering matrix is based on gradients $\frac{\partial \mathbf{I}(\mathbf{x}_i)}{\partial \mathbf{x}}$ in a single pixel $\mathbf{x}_i$ [18]. However, a single pixel estimate makes the steering matrix unstable and prone to noisy perturbation of the data. Therefore, $\mathbf{C}_{\Omega_i}$ is the regularized estimate of the steering matrix which is *averaged* (note the summation in Eq. (1) from a local aggregation of gradients over a rectangular window Ω_i.

As the name suggests, the steering matrix captures the principal directions of local texture from the gradient distribution in the the small neighborhood Ω (mostly 5×5). This idea becomes easy to follow if the spectral decomposition of the steering matrix is brought into picture:

$$\mathbf{C}_{\Omega_i} = \lambda_1 \mathbf{u}_1 \mathbf{u}_1' + \lambda_2 \mathbf{u}_2 \mathbf{u}_2', \quad (2)$$

where, λ_1, λ_2 are the eigenvalues, and $\mathbf{u}_1, \mathbf{u}_2$ are the eignevectors representing principal directions. Denoting singular values as $s_1 = \sqrt{\lambda_1}$ and $s_2 = \sqrt{\lambda_2}$, they are turned into a Riemannian metric by the following regularization (to avoid numerical instabilities) while keeping the eigenvectors unaltered:

$$\mathbf{C}_{\Omega_i} = (s_1 s_2 + \epsilon)^{\alpha} \left(\frac{s_1 + \tau}{s_2 + \tau} \mathbf{u}_1 \mathbf{u}_1' + \frac{s_2 + \tau}{s_1 + \tau} \mathbf{u}_2 \mathbf{u}_2' \right), \quad (3)$$

where ϵ and τ are set at 10^{-1} and 1 respectively, following [18]. The parameter α can be tweaked to boost or suppress the local gradient information depending on the presence of noise. A closed form solution to compute the regularized form of $\mathbf{C}_{\Omega_i}$ as shown in (3) is also included in [18].

Finally, the LSK is defined by the following similarity function between the center pixel $\mathbf{x}_i$ and its surrounding $p \times p$ local neighborhood $\mathbf{x}_j$, normalized as given below,

$$h_{ij} = \frac{\exp(-\Delta \mathbf{x}_{ij}' \mathbf{C}_{\Omega_j} \Delta \mathbf{x}_{ij})}{\sum_j \exp(-\Delta \mathbf{x}_{ij}' \mathbf{C}_{\Omega_j} \Delta \mathbf{x}_{ij})}, \quad j = 1, 2, \ldots, p^2, \quad (4)$$

where $\Delta \mathbf{x}_{ij} = [\mathbf{x}_i - \mathbf{x}_j]'$. The LSK values thus computed at $\mathbf{x}_i$ are concatenated into a $l = p^2$ dimensional vector $\mathbf{h}_i$ as follows: $\mathbf{h}_i = [h_{i1}\ h_{i2}\ \ldots h_{il}] \in \mathbb{R}^l$. Usually $p \times p$ is considered same in size as that of Ω_j and is set at 5×5.

Note multiple sources of motivation exist to arrive at the expression of LSK (4). A detailed treatment is the out of the present scope, but it is worth mentioning that LSK can be motivated and derived from a geodesic interpretation of signal

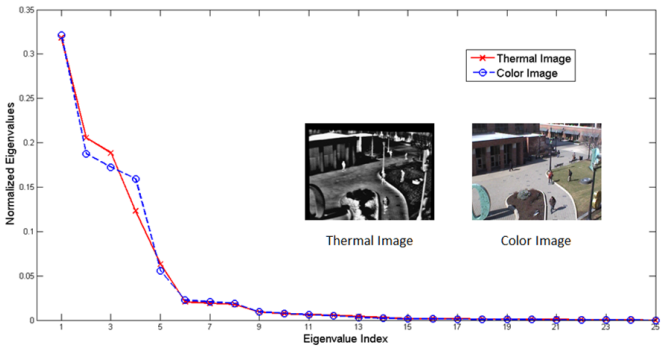


Fig. 3. LSK descriptors belong to a low dimensional manifold where 70% to 80% of the energy of the eigenvalues comes from first three or four.

manifold [20], the kernel view of filtering [29], and definitely from the idea and definition of structure tensor [30].

Irrespective of all the different sources of motivation and derivation the physical significance explaining the functionality of LSK remains plane and simple: aggressively capturing the locally dominant pattern. Thus it comes as no surprise why LSK features can better retain the overall geometry of the signal manifold in contrast to HOG, illustrated in Fig. 2. The comparative visualization in Fig. 2 confirms that the aggregation of local gradients to estimate principal orientation pattern is able to encode the local geometry exceedingly well as compared to the histogram based statistics of HOG.

IV. Decorrelation of Local Descriptors

The descriptor vectors $\mathbf{h}_i$ are stacked together as mode-3 fibers [31], [32] of a third order descriptor tensor $\mathbf{H} \in \mathbb{R}^{M\times N\times l}$. Every i-th channel of $\mathbf{H}$, where $i = 1, 2, \ldots, l$, encodes some directional property of the image $\mathbf{I}$. For example, some channels in $\mathbf{H}$ exhibit vertical structures and some horizontal ones, whereas others capture various oblique directions to a varying degree (Fig. 2 would give a fair idea).

Besides exhibiting the directional characteristics in them the channels are also observed to be sparse. This behavior is directly reflected in Fig. 3 where the spectral decomposition of LSK features $\mathbf{h}_i$ reveals that most of the spectrum energy is stored in the leading few eigenvalues.

In the next step we project the high dimensional tensor $\mathbf{H}$ onto the principal subspaces along its third mode [32], [33]. To be specific, we collect the set of d eigenvectors (computed from LSK descriptors $\mathbf{h}_i$) as columns of $\mathbf{V} = [\mathbf{v}_1\mathbf{v}_2\ldots\mathbf{v}_d] \in \mathbb{R}^{l\times d}$. We choose the number of eigenvalues d in such a way that 80% of the spectral information is contained in the chosen eigenvalues λ as follows: $d = \underset{i}{\operatorname{argmin}} \frac{\lambda_i}{\sum_j \lambda_j} > 80\%$.

Following the *n-mode* product between a higher order tensor and a matrix [33], [34] we compute the 3-mode product (denoted by $\times_3$) between the descriptor tensor $\mathbf{H}$ and subspace $\mathbf{V}$. In other words, the mode-3 (l-dimensional) fibers of $\mathbf{H}$ are projected on the column space of $\mathbf{V}$. As a result, we obtain the feature tensor $\mathbf{F} \in \mathbb{R}^{M\times N\times d}$, where $d << l$, given by the following tensor-matrix product:

$$\mathbf{F} = \mathbf{H} \times_3 \mathbf{V}. \tag{5}$$

Doing this has two imminent benefits: one, the projected descriptors are clean, prominent and discriminating (Fig. 2), and two, reducing the number of feature channels in $\mathbf{F}$ has the runtime benefit for fast detection.

V. Design of Linear Detector with MCS Kernel

In the context of one shot object detection, Seo *et al.* [17] and Biswas *et al.* [20] have shown the effectiveness of Matrix Cosine Similarity (MCS) as a decision rule for computing similarity between two feature tensors (of same size). This measure of image similarity is in fact a generalization of cosine similarity from vector features to matrix/tensor features based on the notion of *Frobenius Inner Product* $\langle\cdot,\cdot\rangle_F$. In principle, suppose $\mathbf{F}_Q \in \mathbb{R}^{m\times n\times d}$ is a query feature tensor that we try to find in a bigger target tensor $\mathbf{F}$ in a sliding window fashion. At each position $\mathbf{x}_i$ of the sliding window over target $\mathbf{F}$ we compute MCS (ρ) as follows:

$$\rho(\mathbf{F}_Q, \mathbf{F}(\mathbf{x}_i)) = \left\langle \frac{\mathbf{F}_Q}{\|\mathbf{F}_Q\|}, \frac{\mathbf{F}(\mathbf{x}_i)}{\|\mathbf{F}(\mathbf{x}_i)\|} \right\rangle_F, \tag{6}$$

where $\|\cdot\|$ denotes *Frobenius norm* for tensors. Higher the value of the MCS at location $\mathbf{x}_i$ in target image, greater is the likelihood of finding the object there. The normalization allows MCS to focus on phase (or angle) information while also taking care of the signal strength. Besides generalizing cosine similarity, this measure also overcomes the inherent disadvantage of conventional Euclidean distance metric which is sensitive to outliers [35]–[37].

It is important to note that MCS also serves as a valid kernel. To show how, it is quite straight forward to write (6) in terms of an inner product between two vectors: $\rho(\mathbf{F}_Q, \mathbf{F}(\mathbf{x}_i)) = \left(\frac{\text{vec}(\mathbf{F}_Q)}{\|\mathbf{F}_Q\|}\right)' \left(\frac{\text{vec}(\mathbf{F}(\mathbf{x}_i))}{\|\mathbf{F}(\mathbf{x}_i)\|}\right)$, where $\text{vec}(\cdot)$ denotes the conventional vectorization operation by stacking the elements of matrix into a long vector. Following this, the proof of MCS being a valid kernel becomes trivial [38], [39].

At its core MCS serves as a cross correlation operator between two tensor signals followed by a normalization of signal strength. Closer inspection reveals that such correlation can be performed separately along each feature channel of the third order tensors. The channel correlations can then be combined by summing them up in the next step. To convey the idea, the *Frobenius Inner Product* of (6) is written below as a multichannel cross correlation,

$$\begin{aligned}
&\rho(\mathbf{F}_Q, \mathbf{F}(\mathbf{x}_i)) \\
&= \sum_d \sum_n \sum_m \frac{\mathbf{F}_Q(m,n,d)}{\|\mathbf{F}_Q\|} . \frac{\mathbf{F}(x_{i1}+m, x_{i2}+n, d)}{\|\mathbf{F}(\mathbf{x}_i)\|}, \qquad (7)\\
&= \frac{\sum_d \sum_n \sum_m \frac{\mathbf{F}_Q(m,n,d)}{\|\mathbf{F}_Q\|} . \mathbf{F}(x_{i1}+m, x_{i2}+n, d)}{\|\mathbf{F}(\mathbf{x}_i)\|}. \qquad (8)
\end{aligned}$$

The quantity $\|\mathbf{F}(\mathbf{x}_i)\|$ can be pulled out of the summation as it happens to be the Frobenius norm of the tensor located at $\mathbf{x}_i$ of the target, and as such, it does not depend on the interaction between $\mathbf{F}_Q$ and $\mathbf{F}(\mathbf{x}_i)$. This form of the kernel as well and the idea expressed above will be used later to facilitate exact acceleration of MCS computation, described in detail in Section VI.

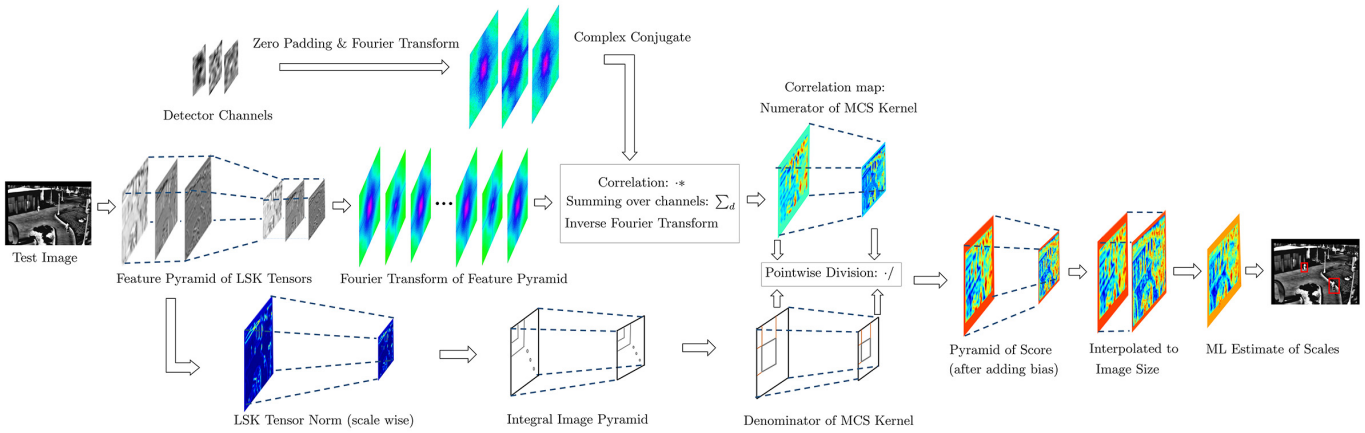


Fig. 4. Multiscale detection technique involving construction of feature pyramid, computation of kernel function and maximum likelihood estimate of scale and location of pedestrian in target image

A. Linear Support Tensor Machine

Following the system overview in Section II, we construct our feature set $\mathbf{F}_i \in \mathbb{R}^{m\times n\times d}$ by first cropping from a full feature tensor (5) according to the ground truth, and next by normalizing with its *Frobenius norm* $\|\mathbf{F}_i\|$. Paying the polite nod to a slight notational abuse we write the final feature tensor in a slightly overloaded form, $\mathbf{F}_i := \frac{\mathbf{F}_i}{\|\mathbf{F}_i\|}$. The learning problem consists of deriving a decision rule based on the set of labeled examples $\mathcal{D} = \{\mathbf{F}_i, y_i\}_{i=1}^N$, where $\mathbf{F}_i$ has the associated label $y_i \in \{+1, -1\}$, representing one of the two classes, i.e., the pedestrian or the background.

We start our maximum margin formulation by noting that the linear classifier in vector space $\mathbb{R}^d$ is represented by $f(\mathbf{a}; \mathbf{w}, b) = \mathbf{a}'\mathbf{w} + b$. A reasonable way to extend such concept of linear classifier from the vector to the tensor space is the following:

$$f(\mathbf{F}; \mathbf{W}, b) = \langle \mathbf{F}, \mathbf{W}\rangle_F + b, \tag{9}$$

where $\mathbf{W} \in \mathbb{R}^{m\times n\times d}$ is a third order tensor template that we aim to learn from the annotated examples $\mathcal{D}$. In general, the resulting optimization that follows to learn $\mathbf{W}$ is given below in its general form:

$$\underset{W,b}{\text{minimize}}\ \frac{1}{2}\|\mathbf{W}\|^2 + C\sum_{i=1}^{N}\mathcal{L}(y_i f(\mathbf{F}_i; \mathbf{W}, b)), \tag{10}$$

where $\mathcal{L}(\cdot)$ denotes the loss over data and $C > 0$ is a tradeoff between regularization and constraint violation. Though not critical in the sense that there exists many losses, we have used hinge loss because of its simplicity and wide usage. Henceforth we would assume, $\mathcal{L}(x) = \max(0, 1 - x)$.

Note the MCS kernel can be expressed as an inner product between two vectors. Hence it leads to positive definite real valued kernel with corresponding Reproducing Kernel Hilbert Space (RKHS). Also, the regularization in (10), in the form of *Frobenius norm* $\|\mathbf{W}\|$, is a generalization of vector norm to tensor space, and thus can be shown monotonically increasing real valued function. Under such circumstances, the solution $\widetilde{\mathbf{W}}$ to (10) can also be written as a linear combination of feature tensors as the direct result of the well known Representer theorem [38], [39]:

$$\widetilde{\mathbf{W}} = \sum_{j=1}^{N} y_j\beta_j\mathbf{F_j}. \tag{11}$$

When we insert (11) in (10) the optimization takes place over $(\beta_1, \beta_2, \ldots, \beta_N) \in \mathbb{R}^N$ in the dual domain instead of $\mathbf{W} \in \mathbb{R}^{(m\cdot n\cdot d)}$ in the primal. Following the optimization, we arrive at the desired classifier function below:

$$f(\mathbf{F}(\mathbf{x}_i); \beta_1, \beta_2, \ldots, \beta_q, \tilde{b}) = \sum_{j=1}^{q} y_j\beta_j\rho(\mathbf{F}_j, \mathbf{F}(\mathbf{x}_i)) + \tilde{b}, \tag{12}$$

where, $\mathbf{F}(\mathbf{x}_i)$ according to our past notation denotes the feature tensor at location $\mathbf{x}_i$ inside $\mathbf{F}$, and $\tilde{b}$ is the minimizer of (10) with respect to b . In short, q (where $q \leq N$) kernel computations are needed to classify a tensor $\mathbf{F}(\mathbf{x}_i)$ using the q support tensors. For high dimensional dataset (where $N << m\cdot n\cdot d$) a dual solver for training is preferred because one optimizes less number of parameters: $\beta_1, \beta_2, \ldots, \beta_N$. However, with the increase in dataset size, especially with a large N, a primal solver (e.g., stochastic gradient descent [2], [40]) offers attractive benefit in terms of simplicity and smaller cache size.

While dealing with tensors it is important to note that multilinear algebra provides an inner glimpse of each mode (i.e., dimension) of the tensor — the roles that they supposedly play (e.g., causal factors like illumination and pose [41]). The idea here is to apply the linear model $\mathbf{a}'\mathbf{w}+b$ but separately in each dimension [42]. This becomes possible by constraining the template tensor $\mathbf{W} \in \mathbb{R}^{m\times n\times d}$ to be a sum of R rank-1 tensors — a direct result of CANDECOMP/PARAFAC (CP) decomposition of higher order tensors. This is given by the following,

$$\mathbf{W} = \sum_{r}^{R} \mathbf{w}_r^{(1)} \circ \mathbf{w}_r^{(2)} \circ \ldots \circ \mathbf{w}_r^{(c)}, \tag{13}$$

where '$\circ$' represents the tensor outer product [32], [34]. Inserting (13) in (9) results in the following form of the classifier:

$$f(\mathbf{F}_i; \mathbf{W}, b) = \langle \mathbf{F}_i, \sum_{r=1}^{R} \mathbf{w}_r^{(1)} \circ \mathbf{w}_r^{(2)} \circ \ldots \circ \mathbf{w}_r^{(c)} \rangle_F + b, \quad (14)$$

$$= \sum_{r=1}^{R} \langle \mathbf{F}_i, \mathbf{w}_r^{(1)} \circ \mathbf{w}_r^{(2)} \circ \ldots \circ \mathbf{w}_r^{(c)} \rangle_F + b. \quad (15)$$

Once all but $\mathbf{w}_k$ is fixed, $f(\mathbf{F}; \mathbf{W}, b)$ becomes the familiar problem of learning the linear classifier $\mathbf{a}'\mathbf{w}_k + b$. In general, the approach for learning all $\mathbf{w}_k$ is to estimate $\mathbf{w}_k$ from a suitable loss function on $\mathcal{D}$ while treating all other $\mathbf{w}_i$, where $i \neq k$, constant. The learning algorithm proceeds by repeating this step iteratively for all k. In the theory of supervised tensor learning this training methodology is known as alternate projection algorithm [42], or more popularly in computer vision literature, as coordinate descent algorithm [43]. It is worth pointing out that similar methodology has recently been used in tensor regression to estimate object poses [34].

It turns out that though multilinear algebra provides a discriminative way to compute different components of $\mathbf{W}$, the complexity of multilinear support tensor machine is even stricter with total number of parameters being $R(m+n+d)$. Such low complexity no doubt guards the solution from potential overfitting, but also ruins the detector much of its discriminatory power, especially when the number of examples goes high.

In our dataset the number of examples N is comparable to, or even greater than, the complexity of $\mathbf{W}$. This comes as no surprise because the pedestrians on an average look small in our dataset. At the same time, a reasonably high number of them, along with an equally good number of challenging background examples, motivate us to use a rigid template tensor with decent enough complexity. It appears that number of parameters available in multilinear support tensor machine would be to few to handle the variations present in the dataset. Of course, a trade-off can be achieved by experimenting with an increasing R in the rank-1 approximation (13) of $\mathbf{W}$. However, it is not clear at this point whether such endeavor is justified in exchange of the much simpler but effective linear model (11)-(12).

The form of classifier (12) provides us further insight in the detector development. To see this, we simplify (12) as follows:

$$f(\mathbf{F}(\mathbf{x}_i); \beta_1, \beta_2, \ldots, \beta_q, \tilde{b})$$

$$= \sum_{j=1}^{q} y_j \beta_j \langle \mathbf{F}_j, \mathbf{F}(\mathbf{x}_i) \rangle_F + \tilde{b}, \quad (16)$$

$$= \langle \sum_{j=1}^{q} y_j \beta_j \mathbf{F}_j, \mathbf{F}(\mathbf{x}_i) \rangle_F + \tilde{b}, \quad (17)$$

$$= \langle \widetilde{\mathbf{W}}, \mathbf{F}(\mathbf{x}_i) \rangle_F + \tilde{b} = f(\mathbf{F}(\mathbf{x}_i); \widetilde{\mathbf{W}}, \tilde{b}), \quad (18)$$

where, $\widetilde{\mathbf{W}} = \sum_{j=1}^{q} y_j \beta_j \mathbf{F}_j$ as a linear combination of q support tensors forms our detector (with a bias b). We are able to write the first step because tensors are normalized and MCS in such case boils down to *Frobenius inner product* (Section V). Second step results by virtue of the linearity of an inner product. In the last step the decision boundary is parameterized in terms of $\mathbf{W}$ instead of α_j.

A few notes follow from our proposed design decisions. First, as also noted by [44], linear support tensor machine makes the training simpler (i.e., one stage) in contrast to multilinear learning, and any off-the-shelf support vector machine solver can handle the optimization problem. Second, even if we are using linear classifiers we deliberately maintain the tensor form of the features. We do not recommend vectorization of tensor features because even if that is permissible for easier training with existing solvers, our detection (prediction) stage would make explicit use of the tensor form for fast and efficient computation. This efficiency resulting from the tensor representation not only aids prediction but also shortens the training time by quickening the hard mining stage.

VI. Exact Acceleration of Tensor Classifier

Searching the detector $\widetilde{\mathbf{W}} \in \mathbb{R}^{(m \cdot n \cdot d)}$ in a bigger target tensor $\mathbf{F} \in \mathbb{R}^{M \times N \times d}$ ($M >> m, N >> n$) by repeated evaluation of the decision rule (18) is a computationally intensive task. For example, a single channel detector (i.e., $d = 1$) of size $m \times n$ when searched in an $M \times N$ target tensor incurs a computational cost $\mathcal{O}(mnMN)$, and with d feature channels this cost becomes $\mathcal{O}(dmnMN)$. Using the classifier computed in (18), we ascertain the scores at each pixel position $\mathbf{x}_i$ of the feature tensor $\mathbf{F}$ as follows,

$$f(\mathbf{F}(\mathbf{x}_i); \widetilde{\mathbf{W}}, \tilde{b})$$

$$= \sum_{d} \sum_{n} \sum_{m} \widetilde{\mathbf{W}}(m,n,d) \cdot \frac{\mathbf{F}(x_{i1}+m, x_{i2}+n, d)}{\|\mathbf{F}(\mathbf{x}_i)\|} + \tilde{b}, \quad (19)$$

$$= \frac{\sum_d \sum_n \sum_m \widetilde{\mathbf{W}}(m,n,d).\mathbf{F}(x_{i1}+m, x_{i2}+n, d)}{\|\mathbf{F}(\mathbf{x}_i)\|} + \tilde{b}, \quad (20)$$

The denominator $\|\mathbf{F}(\mathbf{x}_i)\|$ does not include channel wise interaction with $\widetilde{\mathbf{W}}$ and thus can be taken out of the sum. The numerator in (20) involves channel wise cross-correlation (represented by two inner summation) followed by summation across channels (outermost summation). Cross-correlation, especially for a high ratio in sizes between the detector and target tensors, happens very fast in frequency domain. Therefore, to reduce the runtime computation we precompute the channel wise Fourier transforms of the detector tensor $\widetilde{\mathbf{W}}$.

During runtime, we perform Fourier transform $\mathcal{F}\{\cdot\}$ of each feature channel $\mathbf{F}(:,:,i), \forall i = 1, 2, \ldots, d$, perform point-by-point multiplication in frequency domain, the correlation channels thus obtained are summed up right in frequency domain (owing to the linearity of MCS that remains preserved in Fourier transform), and lastly, we invert back the correlation plane in spatial domain by applying inverse Fourier transform $\mathcal{F}^{-1}\{\cdot\}$. The whole process of computing the numerator in

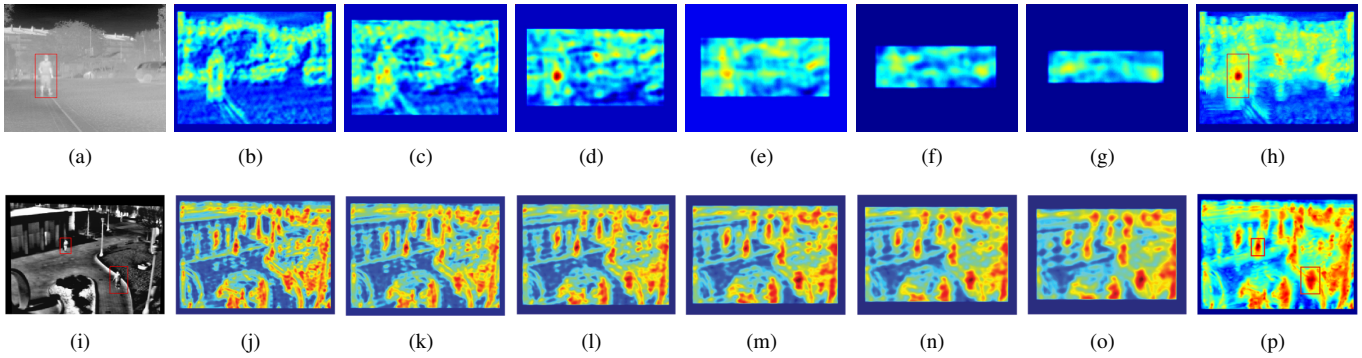


Fig. 5. **Scale Estimation in Multiscale Detection**: Each scale of features in the feature pyramid yields a score map during detection. The individual score maps of various sizes are rescaled with bilinear interpolation to a common size ((b)-(g) or (j)-(o)). Note, the boundary region in the score map is getting wider (filled with zeros) with scales because of the decreasing target size in feature pyramid (Fig. 3). The maximum score at each pixel location is then selected from all score maps to obtain the final score map (h), or (p), which upon thresholding and non-maximal suppression yields pedestrian location. The score map supplying the maximum score at a particular pixel provides the scale index that determines the size of the bounding box. The usual convention of colormap is followed (blue means low score and dark red to reddish black denote high score).

(20), for all locations $\mathbf{x}_i$, is summarized in the following:

$$f(\mathbf{F}(\mathbf{x}_i); \widetilde{\mathbf{W}}, \tilde{b}) = \frac{\left[\mathcal{F}^{-1}\{\sum_d \mathcal{F}\{\mathbf{F}(:,:,d)\} \cdot * \mathcal{F}^{\dagger}\{\widetilde{\mathbf{W}}(:,:,d)\}\}\right]_{\mathbf{x}_i}}{\|\mathbf{F}(\mathbf{x}_i)\|} + \tilde{b}, \tag{21}$$

where $\mathcal{F}^{\dagger}\{\cdot\}$ denotes conjugated Fourier transform [1]. The $[\cdot]_{\mathbf{x}_i}$ in the numerator denotes the correlation score at $\mathbf{x}_i$ that subsequently gets divided by the normalization factor present in the denominator. Thus computing the numerator of $f(\mathbf{F}(\mathbf{x}_i); \widetilde{\mathbf{W}}, b)$ at every pixel location $\mathbf{x}_i$ in the target tensor takes $\mathcal{O}(dMN \log MN)$ for forward as well as inverse Fourier transform, and $\mathcal{O}(dMN)$ for point by point multiplication as well as for summation. Eventually, we end up with an overall time complexity of $\mathcal{O}(dMN \log MN)$.

The last step in (21) involves normalization by $\|\mathbf{F}(\mathbf{x}_i)\|$ which could be performed efficiently by computing an integral image of $\sum_d \mathbf{F}(:,:,d) \cdot * \mathbf{F}(:,:,d)$ involving a time complexity $\mathcal{O}(dMN)$. The retrieval of the normalization value eventually follows from the square root in constant time per window. In summary, the computation cost of the classifier is dominated in the numerator by $\mathcal{O}(dMN \log MN)$, and by $\mathcal{O}(dMN)$ in the denominator, leading to an overall complexity of $\mathcal{O}(dMN \log MN)$. This is reasonably less in contrast to the brute force time complexity of sliding window detection: $\mathcal{O}(dmnMN)$. The fact that the cost no longer relies on the rigid detector's template size results in a substantial gain in efficiency. The complete methodology for evaluating the proposed MCS is illustrated in details in Fig. 4.

Such idea of accelerating the detection process by a multi-channel implementation of Fourier transform has found recent application in [45], as well as in [20]. In [45], the authors did not have to deal with the normalization factor that is present in MCS. Biswas et. al, [20] extended their technique to accelerate the MCS computation. In this paper, we show how the fast kernel computation can further be extended to efficient pedestrian detection following a tensor based maximum margin learning setup. The proposed acceleration of decision rule does not involve any approximation, hence, it remains an exact version with a much shorter detection time.

[1]It is worth noting that correlation happens when one of the two Fourier transforms is conjugated, whereas convolution takes place with the product of two Fourier transforms without any conjugation

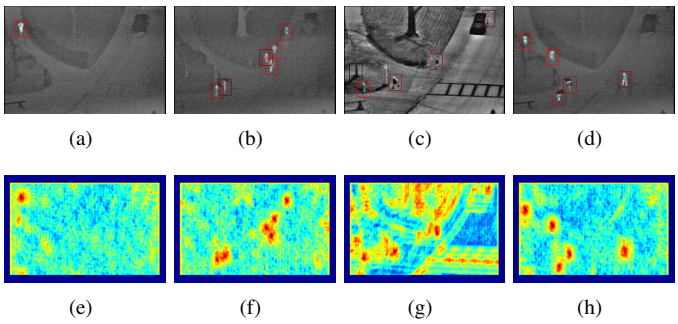


Fig. 6. Detection results of the proposed methodology on OSU-T dataset are shown in this figure. The top row shows the bounding boxes indicating pedestrian location. The bottom row illustrates the scores obtained from the single-scale detector in the form of heat map (the convention of color map is maintained, i.e., red indicates highest confidence and blue lowest.

VII. Multiscale Detection Methodology

There are two approaches generally available for multiscale search of objects. One approach is to scale up the rigid detector and search for maximum scoring region. Though attractive because target image undergoes minimum transformation during runtime, from a purely theoretical standpoint this scaling up of a rigid detector can have the uncanny effect of introducing artifacts in bias b while computing $f(F; \widetilde{\mathbf{W}}, \tilde{b})$ at every location $\mathbf{x}_i$. It is not immediately obvious how and to what extent such issues will manifest in the present methodology, and in case they do, what could be probable way out to mitigate such limitation. Hence, we have followed the second approach that involves target rescaling. To be specific, we computed features from the given image and resorted to feature scaling over the desired range of scales. In other words,

we have constructed a feature pyramid of decreasing image size as described in Fig. 4.

We describe next how we infer the pedestrian's location and the size in the test image. It is important to note that for each scale we essentially obtain a score map as a result of detection (Fig. 5). Each pixel intensity in a particular score map represents the value of scoring function of the proposed linear classifier at that particular scale. We rescale the score maps of all scales to bring them to a common size (the largest scale in our case) before selecting the maximum score at each pixel to best estimate the scale that is producing the maximum detection score. The final score map thus obtained is thresholded (usually at zero) following a non-maximum suppression step to output the pedestrian's location. The maximum scale associated with the pedestrian's location provides the size of the bounding box we need.

VIII. Experiments and Results

Though infrared images often exhibit visual cues that remain absent in the visible spectrum, the clarity and usefulness of such information may be limited by several other extraneous parameters like sensor noise, temperature of objects, weather conditions, indoor and outdoor environments. The extent to which computer vision tasks like visual recognition is influenced by those external factors requires a careful study of large-scale, well annotated infrared datasets that are not still as many, and the size of such datasets, where available, is relatively small [23], [46]. The ready availability of similar useful resources have facilitated steady improvement in the performance of pedestrian detection in natural images (CalTech [3], INRIA pedestrians [1]) over the last few years [5].

To mitigate this shortcoming new and large-scale thermal image datasets have been developed recently, e.g., LSI [25] and KAIST [47] and BU-TIV [48]. The decent image quality of LSI and variable heights of the pedestrians offer a good range of difficulty levels to develop pedestrian detectors. KAIST multispectral images, also captured in a real life setting, shows rapid degradation of image quality in thermal channel with increasing distance, and it gets quite difficult to distinguish distant pedestrians from background with the infrared channel alone (KAIST dataset comes with RGB color channels too). The BU-TIV dataset has relatively high resolution images of human beings for a wide range of visual recognition tasks like detection, single view and multi-view tracking. The objects for the detection task in the BU-TIV not only include pedestrians but also other classes like cars and bikes on a crowded street.

In this work we focus our attention on detecting pedestrians in four baseline datasets: OSU Thermal (OSU-T) [23], OSU Color Thermal (OSU-CT) [24], LSI [25], and KAIST [47]. We restrict our study to thermal channels only. Color channels when available are ignored. The baseline dataset OSU-T contains pedestrians in still images. The other two datasets namely OSU-CT and LSI both are infrared video datasets.

In the experimental setup we have first computed the three dimensional LSK descriptors where third dimension denotes the number of descriptor channels. Such number is always 25 since we have considered 5×5 neighborhood around the central pixel in (4). The number of eigenvectors used for feature computation is typically three unless mentioned otherwise. We have used LIBSVM [49] solver to solve the max-margin optimization task. The choice of a solver is not critical, and the proposed methodology is general enough to solve with any quadratic solver, e.g., [50] and [51] which are available in VLFeat library [52].

For describing the evaluation process we have followed the 50% intersection-over-union PASCAL criterion [53]) between the detected bounding box and the supplied ground truth, to determine correct detection and missed detection (or false negative). In particular, we define the miss rate by $FN/(TP+FN)$, where FN represents the total number of false negatives, and TP the total number of true positives. The total number of false positives or FP is normalized by the number of images in the test set leading to FPPI or false positives per image. Following the evaluation technique established for detecting pedestrians in the visible spectrum [3], we report here the miss rate versus FPPI graph as a measure of detector performance. This is in contrast to earlier notion of false positive per window (FPPW) as used to evaluate the pedestrian detector [1], [7]. Miss rate versus FPPI is also in contrast to the precision-recall curves that are more traditionally followed in other areas of object detections [53]. The present evaluation criterion is motivated by the applications like autonomous driving where it is often the norm to fix the upper ceiling at acceptable FPPI rate independent of the number of pedestrians in the image.

OSU Thermal Database (OSU-T): OSU thermal images [23] come from 10 sequences with a total number of 284 images all of which are 8-bit. This dataset is not a video sequence because the images are captured in a non uniform fashion with a sampling rate less than 30Hz. The image size is 360×240 pixels. In total, the dataset has 984 pedestrians across all 10 sequences.

We have followed a K-fold cross validation technique for the evaluation by holding out each of the 10 sequences for the test, and use the rest of the sequences for training. The detection results of each held-out sequence are later combined in a big text file, and analyzed, to compute the overall miss rate and FPPI [3] for the full dataset. It is worth noting that in the wide area surveillance, like satellite image analysis, it is pretty common to attribute less emphasis to scale. Objects at a distance do not appear to vary widely in sizes. Indeed, with a *single scale rigid detector* we have achieved reasonably good performance as shown in the Fig. 6. Note that the ground truth supplied varies in height and width across pedestrians. However, we have extracted a constant height $\times$ width bounding box (36×28 to be specific) around the center of the given ground truth rectangle for each pedestrian. With the change in weather condition the appearance (illumination in particular) of the background varies significantly. We built an initial detector with a subset of the pedestrian and background tensor features, and collected hard negative examples (like [1], [2]) by trying to detect pedestrians with the initial detector. The α is set to 0.4 in this experiment following the feature extraction setup of [20].

OSU Color Thermal Database (OSU-CT): OSU Color

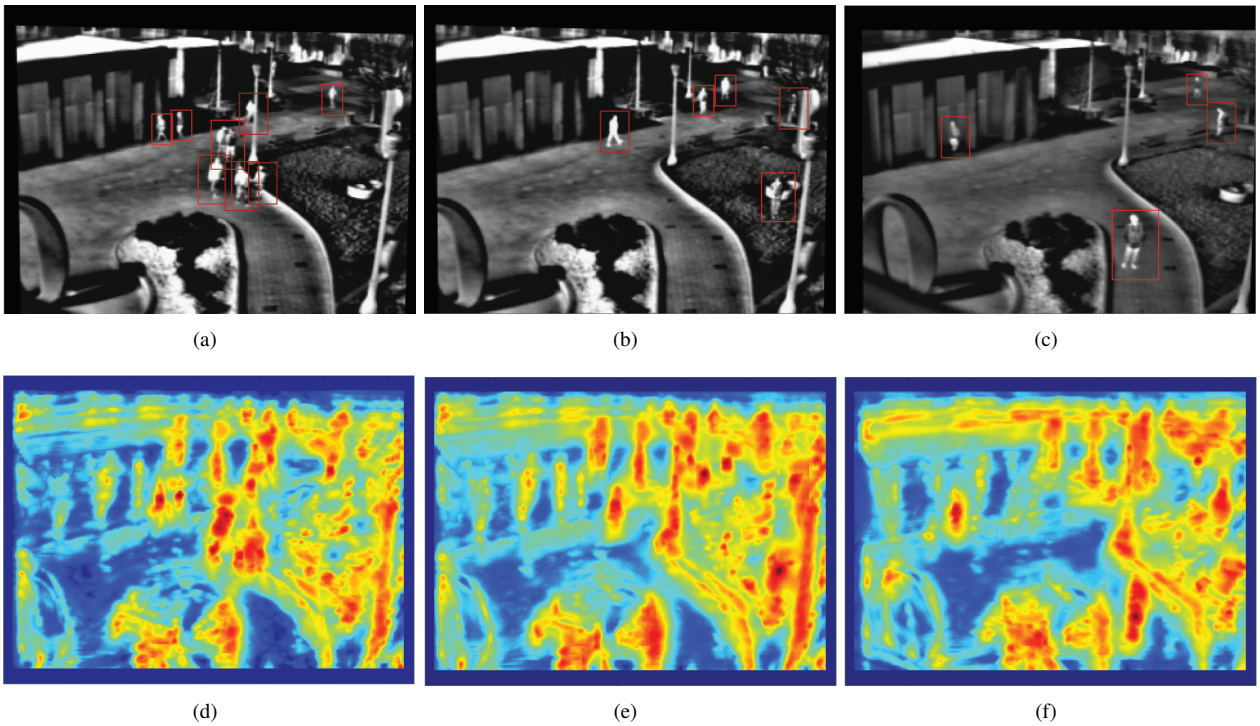


Fig. 7. Top row shows multiscale detection on three frames from OSU-CT dataset. The scale best estimated is shown with the appropriate sized bounding box centered at the predicted location. The bottom row heat maps illustrate corresponding decision scores (maximum likelihood estimate across all six scales) obtained from classifier. The blue regions show less confidence and the red to reddish black shows high to very high confidence in detecting pedestrians. The proposed detector faces difficulty in detecting partially occluded people in absence of any tracking information and/or background model.

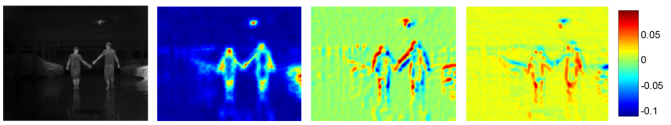


Fig. 8. The thermal image (far left) is shown in three LSK feature channels. Note how the first channel shows signal strength around body silhouette, whereas second channel tends to highlight horizontal to oblique structures. The third channel mostly models the vertical to near vertical structures.

Thermal dataset [24] has a total of 17088 images (8-bit thermal and 24-bit color) and is a video sequence dataset. This dataset has a total of six sequences with each three containing scenes of same location. Each of the six sequences has thermal as well as color channels. Since this dataset does not have a ground truth available in the public domain, we have annotated the full data set (using tools [3], [54], [55]) for the evaluation of the proposed detector. The annotation task gets challenging because the pedestrians at a distance often get occluded by physical structures (e.g., poles and tree branches), or by other pedestrians. In such cases, we have either ignored the heavily occluded pedestrian, or approximately sized the bounding box around a partially occluded person to the best guess possible.

The six thermal sequences have a total number of 8544 images. From a purely pedestrian detection perspective, the last three of the six sequences are somewhat irrelevant because a large number of frames have only one or two pedestrians, and sometimes none. We have experimented with the first three sequences (containing 3355 images in total) that are extremely challenging involving the presence of many pedestrians, heavy occlusions, and low-resolution. Following the sampling procedure of CalTech pedestrian dataset [3], we have uniformly sampled the frames (every 10-th frame) from each of the three video sequences to include in our experiment. In total, we have 1534 pedestrians coming from all the three sequences. Similar to OSU-T we have followed a 3-fold cross validation to complete our evaluation process.

In this dataset, the pedestrians are not as far as those in OSU-T from the camera position. As a consequence, we employ a multi-scale detection strategy to meet our goal. The full dataset contains pedestrians with heights ranging from 14 to 60 pixels. However, we ignore pedestrians which appear too small (less than 20 pixels in height) when they are too far from the camera. We have learnt a rigid detector of size 30 $\times$ 20, and searched it in the target images over the following six scales: 1.30, 1.00, 0.81, 0.68, 0.59, and 0.52. We increase α value to 0.75 to boost the weak singal. An initial detector is computed first to collect hard negative examples from the background of this dataset. We have also applied the initial detector on negative images of LSI to pick hard negatives from this dataset. The final detector is learnt with positive examples from OSU-CT, and hard negative examples from the background of OSU-CT as well as LSI images.

The score maps resulting from the multi-scale detection are shown in Fig. 7 in the form of heat color map. Blue denotes very low confidence, whereas dark red to red-black denotes high to very high confidence in predicting a pedestrian. The heavy occlusion, low-resolution and vertical structures in the

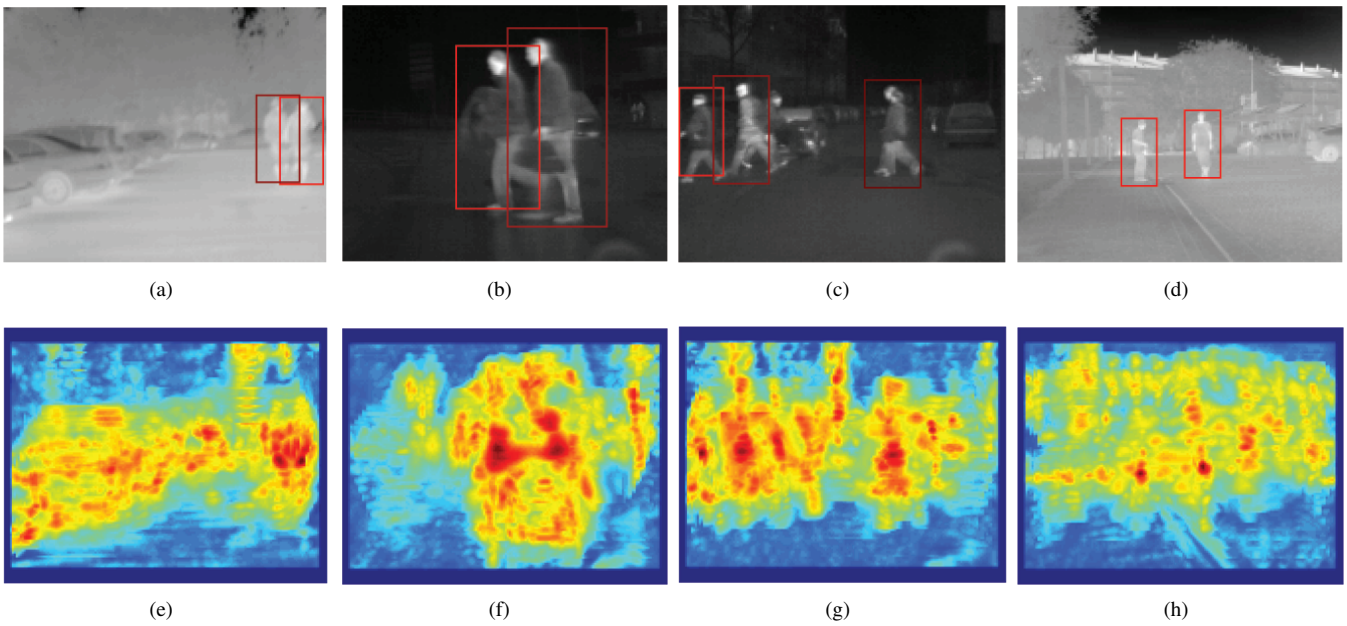


Fig. 9. LSI Results show multiscale detection of pedestrians across wide range of scales. The estimated likelihood of pedestrian's location measured across all the scales is shown under each frame. As before, the dark red to reddish black denotes high to very high confidence of detector.

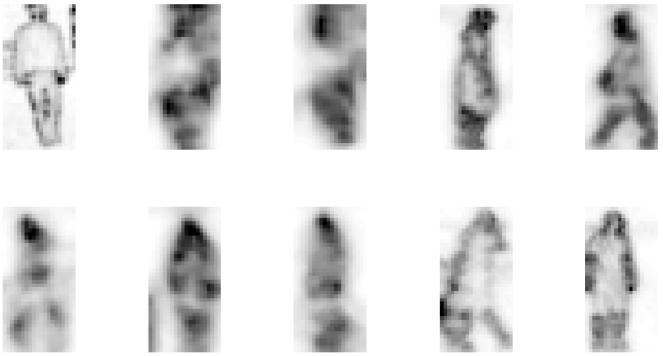


Fig. 10. The shape of pedestrian is prominent positive support tensors shown in the form of first LSK feature channel. More importantly, the positive support tensors show how the linear kernel has succeeded to learn a set of widely different poses of pedestrians.

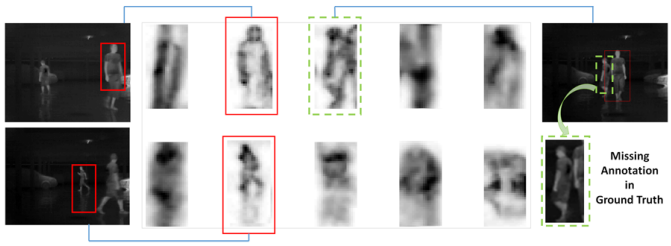
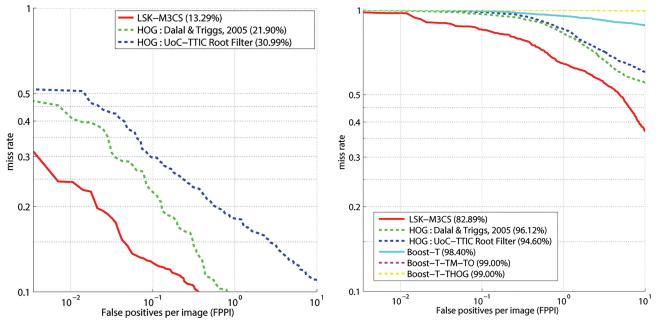


Fig. 11. Negative support tensors are shown to have come from hard mining step where undersized or oversized detection have resulted into false negatives (on left). On right, we show an instance where the correct detection is made but absence of such annotation in ground truth has forced this example into being a false negative.

background make the detection task quite challenging.

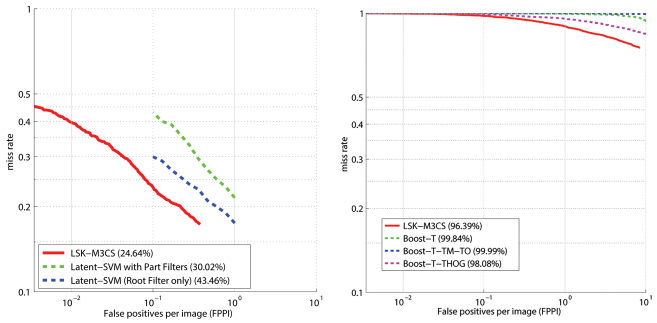
LSI Far Infrared Pedestrian Database: This dataset comes in two flavors, classification setup as well as detection setup. We have focused on the detection set which is further divided in two subsets, training and test set. The training set has 3225 positive images and 1601 negative images. The test set includes 3279 positive and 4859 negative images. The images are 164 pixels wide and 129 pixels tall. Since the intensities of LSI images roughly range from 31000 to 35000 (16 bit images), we have scaled the intensities to 0-255 without noticeable loss in performance. We have followed the usual two-step process for the detector development: building an initial detector in the first step with a subset of positive and randomly sampled negative examples, and in the second step, we consider all positive examples besides including the hard negative examples from the initial detector's output. A tensor of size 40×20 with 3 channels are learnt, and the number of scales in the feature pyramid (Fig. 4) is set at ten, namely, 2.50, 1.58, 1.16, 0.90, 0.75, 0.64, 0.55, 0.49, 0.44, 0.40. We use an The α value of 0.4 in this experiment.

Fig. 8 shows the LSK feature channels corresponding to a thermal image. The LSK features characteristically decomposes the gray scale image in contour, horizontal and vertical segments. In general, we have observed that the number of support tensors in the final model ranges from 15% to 20% of the full dataset. In Fig. 10 we show the positive support tensors by displaying the first channel of LSK tensor feature. One can notice the wide range of poses captured in the learning process of the support tensors. Fig. 11 illustrates negative support tensors which have resulted from hard mining step after being either under or over detected bounding box. The same figure also illustrates an example where missing annotation in ground truth pushes it into hard negative set. This shows that the proposed methodology is robust to noise and outliers present in the ground truth.



(a) (b)

Fig. 12. Miss rate versus false positives per image (FPPI) for the OSU datasets: (a) OSU-T, (b) OSU-CT. The miss rates at 10^{-1} FPPI are mentioned for the proposed LSK-M3CS and other baselines. In case of OSU-CT, we have used thermal (T), gradient magnitude (TM), histogram of oriented gradients (TO) and HOG (HOG) channels with boosting based classifiers (modified ACF [12]) for comparative study.



(a) (b)

Fig. 13. Miss rate versus false positives per image (FPPI) for the other two datasets: (a) LSI thermal dataset, and (b) KAIST multispectral dataset. Like OSU-CT we have modified the ACF detector to work on (only) the thermal images of KAIST dataset.

KAIST Multispectral Database (Thermal channels only): This dataset comes with the sequences of color-thermal image pairs. Following the focus of our paper we use the thermal channels for our study and discard the color channels altogether. The resulting detection task is extremely challenging since thermal images have reasonably high visual ambiguity. The relatively nondescriptive background of thermal images further adds to the difficulty of separating foregrounds from its surroundings.

The authors of this dataset have benefitted from the color information fairly well with the use of ACF detector. Broadly speaking, ACF works by extracting color channels, followed by gradient magnitude and histogram channel computation. All these channels in ACF are fed into a boosting based classifier to learn the pedestrian detector. Since, we do not use color channels, we have modified the ACF detector to make it work solely on the thermal channel. Following the experimental style proposed in [47], we use the following classifiers for the comparative evaluation: only thermal channel (T), thermal channels with gradient magnitude (TM) and histogram of oriented gradients (TO) giving rise to three kind of channel features (T-TM-TO), and lastly thermal and HOG channels computed from thermal image (THOG).

We use the *train20* folder for training and *test* folder for testing the model. In our experiment, we have followed the same parameter setting as available in the released code coming from the authors of the dataset. The height of the humans if less than 55 is ignored. The detector learnt is of dimension 64×32. We have used 12 scales (1, 1.25, 1.33, 1.83, 2.17, 2.50, 2.92, 3.33, 3.75, 4.17, 4.58, 5) for multiscale detection process. As before we have built a initial detector by random sampling of negative windows and positive examples, which when applied to the dataset gives rise to lot of false alarms that are used in augmenting the initial dataset. The augmented dataset is used for deriving the final detector. Fig. 14 shows the results of our detector; all the detections with score greater than zeros are shown. There are plenty of false alarms but it is important to note that such false alarms occupy with low confidence score. the high scores definitely occur in places where human density is high.

A. Results & Discussions

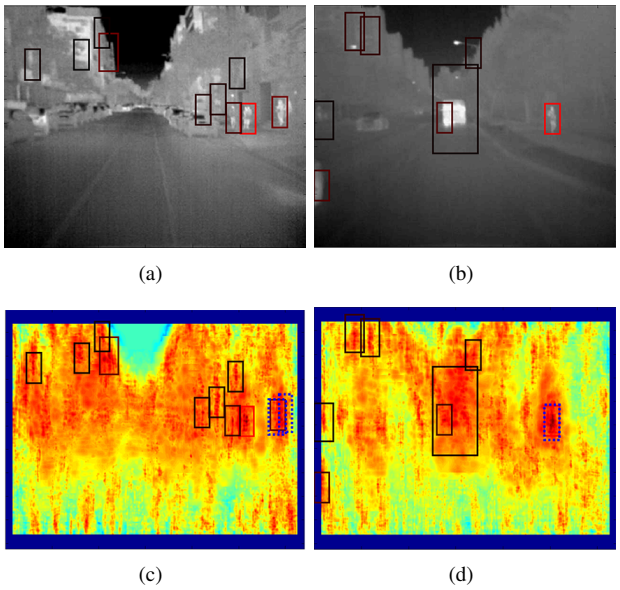
There are usually two approaches available when it comes to computing features, namely, feature engineering and feature learning. Boosting, sparse coding and recently convolutional neural network are learning methodologies one can apply for learning the features from raw image pixels. On the other hand, engineered features, especially HOG has dominated the object detection scenario in the first decade of this century leading to the success of several state of the art detectors, for example, deformable part model.

In our work, however, two of the datasets, namely OSU-T and OSU-CT have pedestrians so small that explicit modeling of parts is not a feasible idea to apply. We have implemented a HOG based linear SVM like [1] using the MATLAB library VLFeat [52] as a baseline for comparison purpose. HOG implemented in VLFeat comes in two forms, one being the originally proposed feature in Dalal and Triggs, 2005 [1], and the other is the dimension-reduced form used in [2] (denoted by UoC-TTIC in Fig. 12). We have compared our methods also with a modification of ACF detectors where the thermal (T), gradient magnitude (TM), and oriented gradients (TO), and HOG computed from thermal channels (THOG) are used to train a boosting classifier [12].

Our proposed LSK with max-margin MCS kernel (LSK-M3SC) detector works superior to HOG based linear SVM both on OSU-T and OSU-CT achieving lowest miss rate (Fig. 12(a) and (b) respectively). However, the extremely occluded nature of pedestrians in OSU-CT has made the detection task challenging for both HOG and LSK with MCS kernel. Fig. 13(a) shows the performance of proposed detector on LSI dataset in comparison with HOG root filter and Latent-SVM with parts [2], [25]. We refer the reader to [25] where the authors have pointed out how introduction of parts in Latent-SVM introduces a derogatory performance on LSI as it often leads to some confusion of the part detector in absence of robust texture in the low resolution and noisy image environment. The proposed feature with our chosen MCS kernel has been able to achieve minimum miss rate as

TABLE I
RUNTIME OF FAST OBJECT DETECTION WITH SCALE ESTIMATION IN COMPARISON WITH SLIDING WINDOW SCHEME

Datasets	Detector Size (pixels)	Image Size (pixels)	Detection time (in seconds) / Frames per second					
			LSK M3CS	HOG (DT)	HOG (UoC-TTIC)	Boosted Channels		
						T	T-TM-TO	T-THOG
OSU-T (single scale)	36 × 28	240× 360	0.15 / 6.67	6.42 / 0.16	7.61 / 0.13	-	-	-
OSU-CT (6 scales)	30 × 20	240× 320 (1.3 to 0.5)	0.28 / 3.57	24.01 / 0.04	26.41 / 0.04	0.01 / 100	0.33 / 3.03	0.32 / 3.13
KAIST (12 scales)	64 ×32	512×640 (1.3 to 0.5)	2.10 / 0.48	-	-	0.09 / 11.11	1.65 / 0.61	0.13 / 7.69



(a) (b) (c) (d)

Fig. 14. Detection results on KAIST Multispectral Dataset: the top row shows detection results with scale estimation. Here, red bounding box denotes a very high confidence score. In the bottom row one can see the score maps. Here, the blue annotations denote ground truth.

shown in Fig. 13(a). In case of KAIST dataset it is worth noting that all the classifiers, including ours, suffer from a high miss rate owing to the noisy nature of the video (Fig. 13(b)). It is worth pointing out that the baseline methods used in the experiment of Fig. 13(b) have exactly the same parameter setting as available in the codes released by [47].

Owing to the efficiency of Fourier transform and integral image the detection process is pretty fast. We have conducted our experiment on a pretty standard Intel Xeon 64-bit desktop machine (CPU E3-1246 v3 @ 3.50GHz) with Ubuntu Linux 14.04 LTS. The performance results in terms of runtime and frames per seconds are shown in Table I (DT [1] and UoC-TTIC [2] are two HOG implementations available in VLFeat). In the current implementation, most of the time is spent by the detector in feature computation. Therefore, we have accelerated the feature computation stage with an elementary C-mexfile implementation. The other modules (e.g., non-maximum suppression, scale estimation) of the proposed detection algorithm are implemented in MATLAB. It goes without saying that the HOG based template detection over six scales takes considerably longer time spanning few seconds to complete. The proposed fast detector also accelerates the training process by making the hard mining step quicker. The search of pedestrians over six scales in a typical 240× 360 (in OSU-T dataset) and 240 × 320 image (in OSU-CT dataset) happens in about a second. Because of the large size of the KAIST images the runtime tends to become longer. The boosted channel classifiers seem to work fastest among all for the detection purpose.

It seems all the detectors evaluated, including ours, suffer from heavy occlusion which is a typical characteristic of the OSU-CT and KAIST dataset. To mitigate this limitation and improve the detection performance one either needs explicit occlusion modeling stage, or tracking methodology to reliably solve the data association problem. We mention these directions as probable opportunities for future research.

IX. CONCLUSION

In this paper we have extended and investigated the use of LSK tensors for pedestrian detection task in thermal infrared images. We have argued that when viewed in the lens of tensors, LSK offers many notable advantages like robustness, noise modeling, superior localization performance, and efficient detection. Continuing in this direction we have proposed a general framework for learning a tensor detector with Matrix Cosine Similarity, as a kernel function. The resulting maximum margin framework is able to distinguish pedestrians from background in challenging scenarios ranging from low signal images to detection at a far away distance. An exact acceleration of the classifier function is proposed by leveraging the tensor form of features as well as multi-channel signal processing techniques. The proposed methodology is compared with other state of the art detectors known to perform well with visible range sensors on the publicly available data sets of thermal infrared images.

REFERENCES

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.
[2] P. Felzenszwalb, R. 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.
[3] P. Dollar and P. P. C. Wojek, B. Schiele, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.
[4] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2056–2063.

[5] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *European Conference on Computer Vision*. Springer, 2014, pp. 613–627.

[6] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010.

[7] M. Teutsch, T. Mueller, M. Huber, and J. Beyerer, "Low resolution person detection with a moving thermal infrared camera by hot spot classification," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, June 2014, pp. 209–216.

[8] R. Brehar and S. Nedevschi, "Pedestrian detection in infrared images using hog, lbp, gradient magnitude and intensity feature channels," in *IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 1669–1674.

[9] J. Li, W. Gong, W. Li, and X. Liu, "Robust pedestrian detection in thermal infrared imagery using the wavelet transform," *Infrared Physics & Technology*, vol. 53, no. 4, pp. 267–273, 2010.

[10] R. Gade and T. B. Moeslund, "Thermal cameras and applications: a survey," *Machine vision and applications*, vol. 25, no. 1, pp. 245–262, 2014.

[11] T. R. Goodall, A. C. Bovik, and N. G. Paulter, "Tasking on natural statistics of infrared images," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 65–79, 2016.

[12] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, Aug 2014.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[14] R. Benenson, M. Mathias, T. Tuytelaars, and L. V. Gool, "Seeking the strongest rigid detector," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 3666–3673.

[15] N. J. W. Morris, S. Avidan, W. Matusik, and H. Pfister, "Statistics of infrared images," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–7.

[16] 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.

[17] 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.

[18] 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.

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

[20] S. K. Biswas and P. Milanfar, "One shot detection with laplacian object and fast matrix cosine similarity," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 3, pp. 546–562, 2016.

[21] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: Object localization by efficient subwindow search," in *IEEE Conf. on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[22] 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, Dec 2009.

[23] J. W. Davis and M. A. Keck, "A two-stage template approach to person detection in thermal imagery." in *Proc. Workshop on Applications of Computer Vision*. IEEE, 2005.

[24] J. W. Davis and V. Sharma, "Background-subtraction using contour-based fusion of thermal and visible imagery," *Computer Vision and Image Understanding*, vol. 106, no. 2, pp. 162–182, 2007.

[25] D. Olmeda, C. Premebida, U. Nunes, J. Armingol, and A. Escalera, "LSI far infrared pedestrian dataset," 2013. [Online]. Available: http://e-archivo.uc3m.es/handle/10016/17370

[26] 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.

[27] O. Zoidi, A. Tefas, and I. Pitas, "Visual object tracking based on local steering kernels and color histograms," *IEEE Transactions on Circuits and Systems for video technology*, vol. 23, no. 5, pp. 870–882, 2013.

[28] X. You, Q. Li, D. Tao, W. Ou, and M. Gong, "Local metric learning for exemplar-based object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 8, pp. 1265–1276, 2014.

[29] P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 106–128, 2013.

[30] G. Peyré, M. Péchaud, R. Keriven, and L. D. Cohen, "Geodesic methods in computer vision and graphics," *Foundations and Trends® in Computer Graphics and Vision*, vol. 5, no. 3–4, pp. 197–397, 2010.

[31] B. W. Bader and T. G. Kolda, "Algorithm 862: MATLAB tensor classes for fast algorithm prototyping," *ACM Transactions on Mathematical Software*, vol. 32, no. 4, pp. 635–653, December 2006.

[32] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[33] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognition*, vol. 44, no. 7, pp. 1540–1551, 2011.

[34] W. Guo, I. Kotsia, and I. Patras, "Tensor learning for regression," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 816–827, 2012.

[35] Y. Fu, S. Yan, and T. S. Huang, "Correlation metric for generalized feature extraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2229–2235, 2008.

[36] Y. Fu and T. S. Huang, "Image classification using correlation tensor analysis," *IEEE Transactions on Image Processing*, vol. 17, no. 2, pp. 226–234, 2008.

[37] Y. Ma, S. Lao, E. Takikawa, and M. Kawade, "Discriminant analysis in correlation similarity measure space," in *International Conference on Machine Learning*. ACM, 2007, pp. 577–584.

[38] C. H. Lampert and C. Lampert, *Kernel methods in computer vision*. Now Publishers Inc, 2009.

[39] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[40] L. Bottou, "Stochastic gradient descent tricks," in *Neural Networks: Tricks of the Trade*. Springer, 2012, pp. 421–436.

[41] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," in *European Conference on Computer Vision*. Springer, 2002, pp. 447–460.

[42] D. Tao, X. Li, X. Wu, W. Hu, and S. J. Maybank, "Supervised tensor learning," *Knowledge and information systems*, vol. 13, no. 1, pp. 1–42, 2007.

[43] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Bilinear classifiers for visual recognition," in *Advances in Neural Information Processing Systems*, 2009, pp. 1482–1490.

[44] Z. Hao, L. He, B. Chen, and X. Yang, "A linear support higher-order tensor machine for classification," *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2911–2920, 2013.

[45] C. Dubout and F. Fleuret, "Exact acceleration of linear object detectors," in *Proc. European Conference on Computer Vision (ECCV)*. Springer, 2012, pp. 301–311.

[46] R. Miezianko, "Terravic research infrared database," in *IEEE OTCBVS Workshop Series Bench*, 2006.

[47] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1037–1045.

[48] Z. Wu, N. Fuller, D. Theriault, and M. Betke, "A thermal infrared video benchmark for visual analysis," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 201–208.

[49] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[50] S. Shalev-Shwartz and T. Zhang, "Stochastic dual coordinate ascent methods for regularized loss minimization," *Journal of Machine Learning Research*, vol. 14, no. Feb, pp. 567–599, 2013.

[51] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for svm," *Mathematical programming*, vol. 127, no. 1, pp. 3–30, 2011.

[52] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

[53] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[54] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *CVPR*, June 2009.

[55] P. Dollár, "Piotr's Computer Vision Matlab Toolbox (PMT)," https://github.com/pdollar/toolbox.