

Robust Object Tracking Using A Spatial Pyramid Heat Kernel Structural Information Representation

Xi Li^{1a}, Weiming Hu^{2a}, Hanzi Wang^{3b}, Zhongfei Zhang^{4c}

^a*National Laboratory of Pattern Recognition, CASIA, Beijing, China*

^b*University of Adelaide, Australia*

^c*State University of New York, Binghamton, NY 13902, USA*

Abstract

In this paper, we propose an object tracking framework based on a spatial pyramid heat kernel structural information representation. In the tracking framework, we take advantage of heat kernel structural information (HKSI) matrices to represent object appearance, because HKSI matrices perform well in characterizing the edge flow (or structural) information on the object appearance graph. To further capture the multi-level spatial layout information of the HKSI matrices, a spatial pyramid division strategy is adopted. Then, multi-scale HKSI subspace analysis is applied to each spatial pyramid grid at different levels. As a result, several grid-specific HKSI subspace models are obtained and updated by the incremental PCA algorithm. Based on the multi-scale grid-specific HKSI subspace models, we propose a tracking framework using a particle filter to propagate sample distributions over time. Theoretical analysis and experimental evaluations demonstrate the effectiveness of the proposed tracking framework.

Keywords: Object tracking, visual surveillance, appearance modeling

1. Introduction

For visual tracking, handling appearance variations of an object is a fundamental and challenging task. Much work has been done in the domain of modeling object appearance variations.

¹Xi Li has moved to CNRS, TELECOM ParisTech, France. Email: xi-li@telecom-paristech.fr

²Email: wmhu@nlpr.ia.ac.cn

³Email: Hanzi.Wang@ieee.org

⁴Email: zhongfei@cs.binghamton.edu

Hager and Belhumeur [1] propose a tracking algorithm which uses an extended gradient-based optical flow method to handle object tracking under varying illumination conditions. However, the problem with this algorithm is the sensitivity to partial occlusion and noise. In [2], curves or splines are exploited to represent the appearance of an object to develop the Condensation algorithm for contour-based tracking. Due to the simplistic representation scheme, the algorithm is sensitive to pose or illumination changes, resulting in tracking failures under a varying lighting condition. Zhao *et al.* [3] present a fast differential EMD tracking method (DEMD) which is robust to illumination changes. But DEMD only considers the issue of color distribution matching without modeling the appearance changes and capturing the structural information of an object. Silveira and Malis [4] present a new algorithm for handling generic lighting changes. Yet, the algorithm performs poorly in capturing the object's shape information which is important to object tracking in complex scenarios.

Black *et al.* [5] employ a mixture model to represent and recover the appearance changes in consecutive frames. Jepson *et al.* [6] develop a more elaborate mixture model with an online EM algorithm to explicitly model appearance changes during tracking. Zhou *et al.* [7] embed appearance-adaptive models into a particle filter to achieve a robust visual tracking. Wang *et al.* [8] present an adaptive appearance model based on the Gaussian mixture model (GMM) in a joint spatial-color space (referred to as SMOG). SMOG captures rich spatial layout and color information. Yilmaz [9] proposes an object tracking algorithm based on the asymmetric kernel mean shift with adaptively varying the scale and orientation of the kernel. Nguyen *et al.* [10] propose a kernel-based tracking approach using maximum likelihood estimation. Yu *et al.* [11] propose a spatial-appearance model which captures non-rigid appearance variations and recovers all motion parameters efficiently. Li *et al.* [12] use a generalized geometric transform to handle the deformation, articulation, and occlusion of appearance. Ilic and Fua [13] present a non-linear beam model for tracking large deformations. Tran and Davis [14] propose robust regional affine invariant image features for visual tracking. Grabner *et al.* [15] develop a keypoint matching-based tracking method by online learning classifier-based keypoint descriptions. The common problem with the above tracking methods is that they perform poorly in characterizing both local and global interactions among pixels, which is crucial for robust visual tracking under bad conditions.

Lee and Kriegman [16] present an online learning algorithm to incrementally learn a generic appearance model for video-based recognition and tracking. The limitation of this algorithm is to heavily rely on a generic prior model, without

which visual tracking cannot be implemented. Lim *et al.* [17] present a human tracking framework using robust system dynamics identification and nonlinear dimension reduction techniques.

The limitation is that its computational cost is expensive. Black *et al.* [18] present a subspace learning based tracking algorithm with the subspace constancy assumption. A pre-trained, view-based eigenbasis representation is used for modeling appearance variations. However, the algorithm does not work well in the cluttered scene with a large lighting change due to the subspace constancy assumption. Ho *et al.* [19] present a visual tracking algorithm based on linear subspace learning. In order to make subspace learning more efficient, two incremental PCA algorithms are proposed in [20] and [21], respectively. Limy *et al.* [22] propose a generalized tracking framework based on the incremental image-as-vector subspace learning methods with a sample mean update. The common limitation in [19, 20, 22] is that they ignore the spatial layout information of object appearance. To address this problem, Li *et al.* [23] present a visual tracking framework based on online tensor decomposition. The framework relies on image-as-matrix techniques for considering the spatial layout information. Besides, Porikli *et al.* [24] utilize several covariance matrices of image features to capture the spatial correlation information of object appearance. They utilize the affine-invariant Riemannian metric to make some basic statistics on the covariance matrices. Similarly, Li *et al.* [25, 26] propose two appearance models under the Log-Euclidean Riemannian metric. In these two models, the Log-Euclidean covariance matrices of image features are used to represent object appearance. In this way, the local self-correlation information of object appearance is taken into account. Wu *et al.* [29] propose a tracking approach which is capable of incrementally learning a low-dimensional covariance tensor representation. However, the tracking methods [23, 24, 25, 26, 29] have a problem that their appearance models lack a competent object description criterion that captures the intrinsic structural properties of object appearance. Babenko *et al.* [30] present a tracking system based on online multiple instance learning. This system is able to update the appearance model with a set of image patches, which do not need to precisely capture the object of interest. But the limitation of this system is to use Haar-like image features for object representation, which is sensitive to complex appearance variations.

In this paper, we propose a tracking framework based on heat kernel structural information (HKSI) matrices. The HKSI matrices essentially reflect the edge flow (or structural) information of the object appearance graph as heat diffusion time progresses. Using the edge flow (or structural) information, the intrinsic properties of object appearance changes can be precisely captured. The main con-

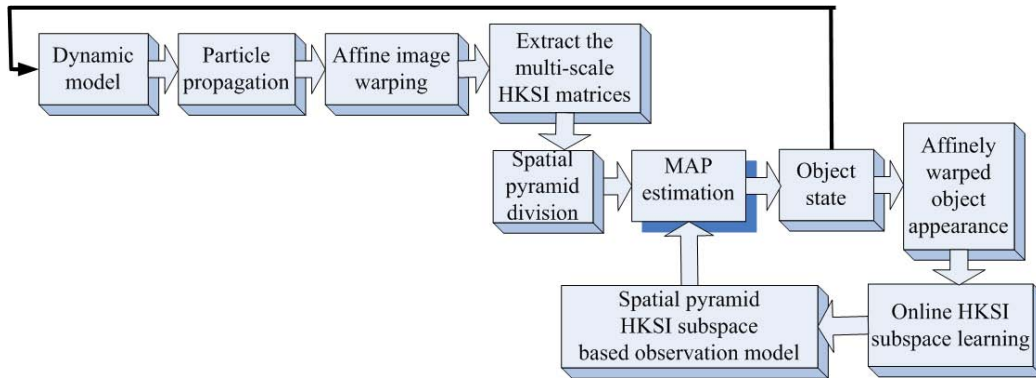


Figure 1: **The architecture of the tracking framework.**

tribution of the tracking framework is summarized as follows. First, an object is represented as an object appearance graph, where a series of multi-scale heat kernel structural information (HKSI) matrices are extracted. These HKSI matrices are capable of capturing the edge flow (or structural) information of object appearance. Second, the spatial pyramid division mechanism is adopted for characterizing the multi-level spatial layout information of the HKSI matrices. Third, a grid-specific HKSI subspace model for each pyramid grid is learned online by the incremental PCA algorithm [22]. Subsequently, the grid-specific HKSI subspace model, which serves as an observation model, is incorporated into a particle filter for tracking. Moreover, a novel criterion for the likelihood evaluation, based on multi-scale HKSI subspace reconstruction error norms, has been proposed to measure the similarity between the test image and the learned grid-specific HKSI subspace model during tracking.

2. The tracking framework

2.1. Overview of the tracking framework

The proposed tracking framework includes three stages: (a) object representation using heat kernel structural information (HKSI) matrices; (b) spatial pyramid division; and (c) Bayesian tracking state inference. In (a), an object region is represented as a collection of heat kernel structural information (HKSI) matrices at different heat diffusion scales. Each HKSI matrix captures the scale-specific structural information of heat diffusion on the object appearance graph. In (b), the HKSI matrix is further divided in a spatial pyramid way. As a result, the

spatial layout information of the HKSI matrix can be effectively captured at different levels. Subsequently, each grid at any spatial pyramid level is represented by a grid-specific HKSI subspace model. In the presence of new frames, the grid-specific HKSI subspace model is online updated by the incremental PCA algorithm [22]. In (c), the object locations in consecutive frames are obtained by maximum a posterior (MAP) estimation within the Bayesian state inference framework in which a particle filter is applied to propagate sample distributions over time. The aforementioned three stages are executed iteratively as time progresses. The architecture of the framework is shown in Fig. 1.

2.2. Object representation using HKSI matrices

In the tracking framework, we construct an object representation based on heat kernel matrices, which can be generated from the following heat equation:

$$\frac{\partial H_t}{\partial t} = -\widehat{L}H_t, \quad (1)$$

where t is a heat diffusion scaling factor controlling the rate of heat diffusion, H_t is the heat kernel matrix, and \widehat{L} is the normalized graph Laplacian matrix.

For convenience, let $\{H_{t_1}, H_{t_2}, \dots, H_{t_K}\}$ denote the heat kernel scale space consisting of K heat kernel diffusion scales. In this case, object representation is governed by a heat kernel scale space. We call H_{t_k} as the scale- t_k heat kernel matrix for $1 \leq k \leq K$. When t_k is larger, H_{t_k} captures more global structural information of object appearance; when t_k is smaller, H_{t_k} captures more local structural information of object appearance. As illustrated in [27, 28], these heat kernel matrices essentially characterize the information flow along the edges of the graph as heat diffusion time progresses. The normalized graph Laplacian determines the rate of flow. The edge flow information corresponds to the intrinsic structural information. Heat kernel matrices at different heat diffusion scales contain the multi-scale edge flow (or structural) information. Thus, we use multi-scale heat kernel matrices for object representation. More theoretical analysis of heat kernel matrices can be found in [27, 28].

The following is the specific procedure of constructing the heat kernel scale space for a given object $Q \in \mathcal{R}^{m \times n}$. The procedure consists of three steps—*object appearance graph creation*, *object appearance graph Laplacian computation*, and *heat kernel mapping*.

- **Object appearance graph creation.** Create a weighted graph with no self-loops $\mathbb{G} = (\mathbb{V}, \mathbb{E}, W)$, where $\mathbb{V} = \{1, \dots, N\}$ is the node set ($N = m \cdot n$

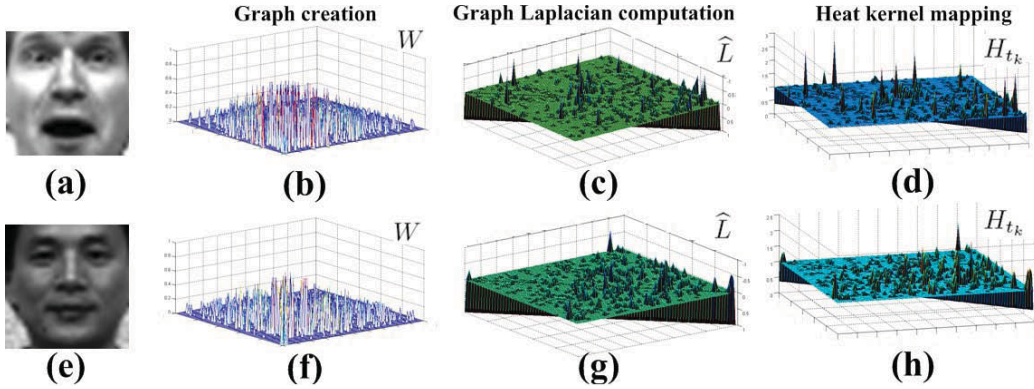


Figure 2: Example of constructing the scale-0.1 heat kernel matrix. (a) and (e) show two different face images; (b) and (f) plot the corresponding edge-weight matrices in the 3D space; (c) and (g) display the corresponding normalized graph Laplacian matrices in the 3D space; (d) and (h) exhibit the corresponding scale-0.1 heat kernel matrices in the 3D space.

is the total number of pixels in $Q \in \mathcal{R}^{m \times n}$, $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ represents the edge set, and $W = (w_{ij})_{N \times N}$ denotes an affinity matrix with the element w_{ij} being the edge weight between nodes i and j :

$$w_{ij} = \begin{cases} \exp\left(-\frac{\|p_i - p_j\|_F^2}{2\sigma_p^2} - \frac{\|c_i - c_j\|_F^2}{2\sigma_c^2}\right) & \text{if } i \neq j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

in which σ_p and σ_c are two scaling factors. More specifically, $p_k = (x_k, y_k)$ is the pixel location, and $c_k = (\mathbb{I}_l^k)_{l=1 \dots \mathcal{L}}$ where \mathbb{I}_l^k is the intensity value of the l -th color channel ($1 \leq k \leq N$), and \mathcal{L} is the number of color channels.

- **Object appearance graph Laplacian computation.** Obtain $L = D - W$ where D is the diagonal matrix with the i th diagonal element being $d_{ii} = \sum_j w_{ij}$ for $1 \leq i \leq N$. Then, transform L into the normalized graph Laplacian $\hat{L} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I_N - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where I_N is an $N \times N$ identity matrix.
- **Heat kernel mapping.** First, define K heat diffusion scales corresponding to a sequence of different values of the scaling factor “ t ” in Eq. (1), i.e., $\mathbb{T} = \{t_1, \dots, t_K\}$. The goal is to capture the edge flow information at different rates of heat diffusion in Eq. (1) on the object appearance graph. Then, perform the spectral decomposition of the normalized graph Laplacian $\hat{L} = \Phi \Lambda \Phi^T$, where Φ and Λ are the eigenvector and eigenvalue matrices, respectively. Finally, compute the heat kernel $H_{t_k} = \exp(-t_k \hat{L}) =$

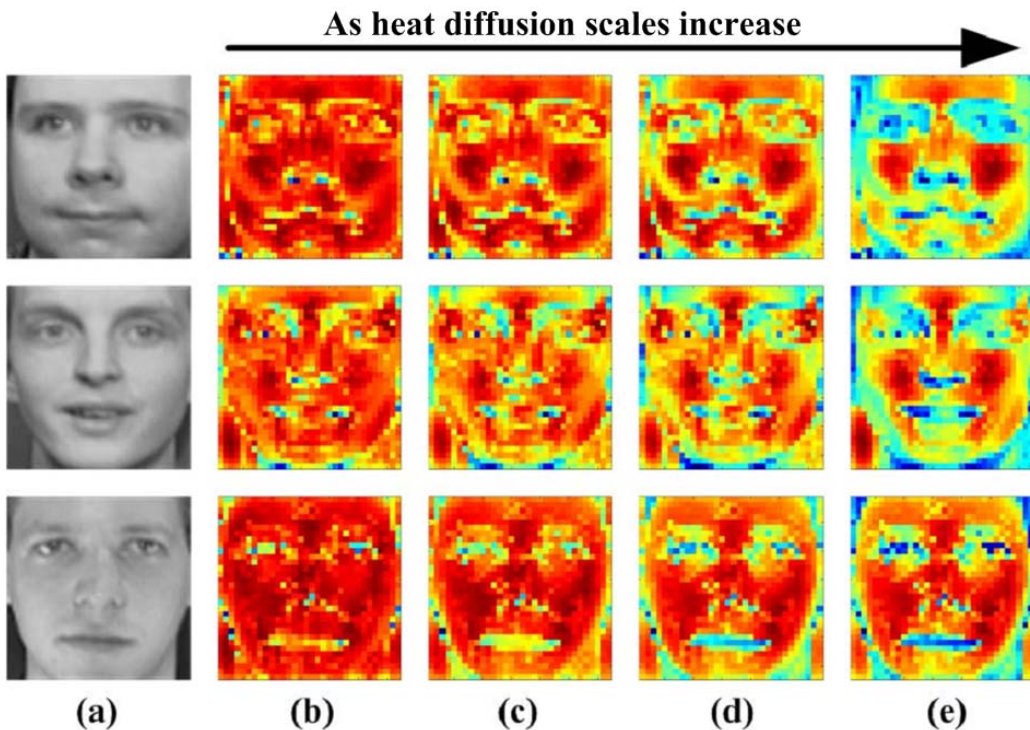


Figure 3: **Illustration of the multi-scale heat kernel structural information matrices.** (a) shows three different face images; (b)-(e) display the corresponding heat kernel structural information matrices at four different heat diffusion scales, respectively.

$\Phi \exp(-t_k \Lambda) \Phi^T$ for $1 \leq k \leq K$ and $H_{t_k} \in \mathcal{R}^{N \times N}$. As a result, we obtain the heat kernel scale space $\{H_{t_1}, \dots, H_{t_K}\}$.

Fig. 2 gives an example of constructing the scale-0.1 heat kernel matrix (i.e., $t_k = 0.1$ for H_{t_k} , and the reason for choosing t_k to 0.1 is that the experimental performance is good when $t_k = 0.1$). More specifically, Figs. 2 (b) and (f) show the similar edge-weight matrices of two different faces in the 3D space. Moreover, Figs. 2 (c) and (g) plot the corresponding normalized Laplacian matrices in the 3D space. From Figs. 2 (c) and (g), we can see that the edge-weight responses of two normalized Laplacian matrices are not significantly different (i.e., very few high peak points). However in Figs. 2 (d) and (h), there are several significant differences (i.e., many high peak points) between two heat kernel matrices at the heat diffusion scale 0.1 (please see the peak points of Fig. 2 (d) for details). Since the dimensions of $H_{t_k} \in \mathcal{R}^{N \times N}$ are usually high, the computational and memory costs are expensive. In order to efficiently mine the structural information

of object appearance, we introduce a scale- t_k heat kernel structural information (HKSI) matrix $\mathcal{S}_{t_k} \in \mathcal{R}^{m \times n}$, which is obtained by summing up each row of the scale- t_k heat kernel matrix $H_{t_k} \in \mathcal{R}^{N \times N}$ into a column vector, and then folding the column vector into an $m \times n$ matrix with the same dimensions as the given object $\mathcal{Q} \in \mathcal{R}^{m \times n}$. The resulting scale- t_k HKSI matrix \mathcal{S}_{t_k} approximately reflects the intrinsic structural properties of object appearance. Considering K heat diffusion scales, we have K HKSI matrices denoted as $\{\mathcal{S}_{t_k}\}_{k=1}^K$. Fig. 3 shows that HKSI matrices at different heat diffusion scales characterize the full spectrum of the intrinsic structural information of face appearance.

In the next subsection, we will discuss online HKSI subspace learning for HKSI matrices. The incremental PCA algorithm [22] is used to learn a multi-scale heat kernel structural information subspace model for HKSI matrices. Since the incremental PCA algorithm takes an image-as-vector representation, each HKSI matrix should be unfolded into its corresponding column vector before subspace learning.

2.3. Online HKSI subspace learning

Given a collection of scale- t_k HKSI matrices $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$, we aim to identify the dominant projection subspace (i.e., U_{t_k} and $\bar{\mathcal{S}}_{t_k}$) of $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$. More specifically, U_{t_k} and $\bar{\mathcal{S}}_{t_k}$ denote the corresponding dominant projection matrix and the mean HKSI matrix of $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$, respectively. During tracking, a number of new scale- t_k heat kernel matrices are added to $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$, resulting in the extension of $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$ along \mathcal{I} . Consequently, we need to identify the dominant projection subspace of $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$ in an online way. Based on the incremental PCA algorithm [22], the scale- t_k heat kernel subspace (i.e., U_{t_k} and $\bar{\mathcal{S}}_{t_k}$) can be efficiently computed online. After computing the HKSI subspaces at different heat diffusion scales, we obtain the multi-scale HKSI subspace model denoted as $\mathcal{E} = \{(U_{t_1}, \bar{\mathcal{S}}_{t_1}), \dots, (U_{t_K}, \bar{\mathcal{S}}_{t_K})\}$.

Furthermore, it is necessary for a subspace learning based algorithm to evaluate the likelihood between the test sample and the learned subspace. In our tracking framework, the criterion for the likelihood evaluation is given as follows. Given the corresponding multi-scale HKSI matrices (i.e., $\{\mathcal{S}_{t_1}, \mathcal{S}_{t_2}, \dots, \mathcal{S}_{t_K}\}$) of a test sample \mathcal{T} , and the learned multi-scale HKSI subspace models (i.e., $\mathcal{E} = \{(U_{t_1}, \bar{\mathcal{S}}_{t_1}), \dots, (U_{t_K}, \bar{\mathcal{S}}_{t_K})\}$), the likelihood can be determined by the sum of the reconstruction error norms:

$$RE = \frac{1}{K} \sum_{k=1}^K \|\text{UM}(\mathcal{S}_{t_k} - \bar{\mathcal{S}}_{t_k}) - U_{t_k}(U_{t_k})^T \text{UM}(\mathcal{S}_{t_k} - \bar{\mathcal{S}}_{t_k})\|_F^2, \quad (3)$$

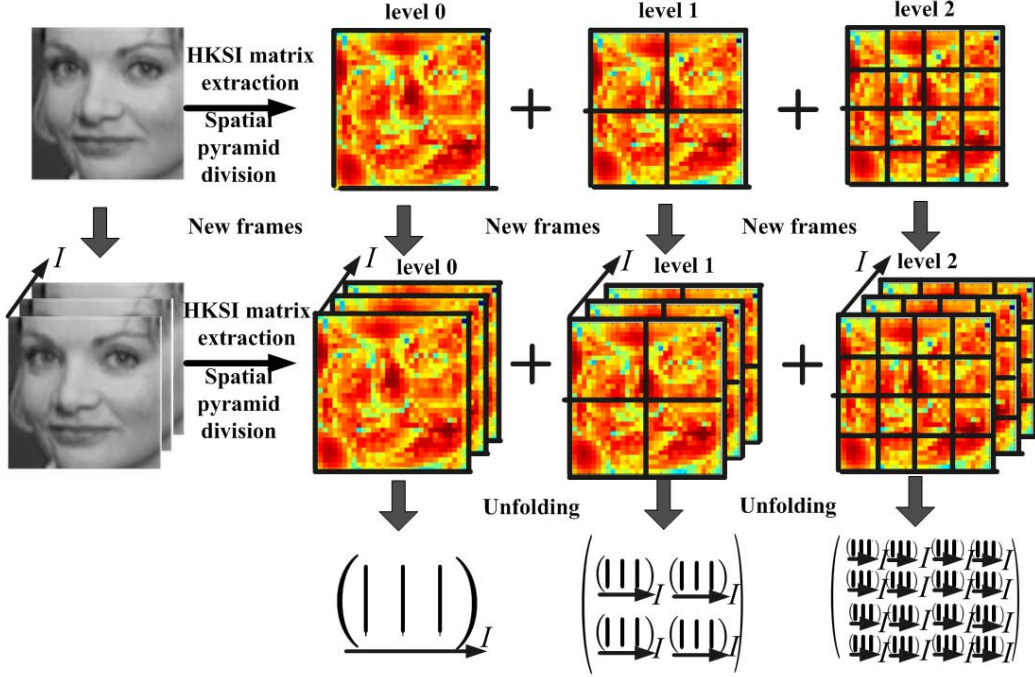


Figure 4: **Illustration of multi-level spatial pyramid division.**

where $\text{UM}(\cdot)$ is an operator unfolding a matrix into a column vector.

In order to further characterize the spatial layout information of HKSI matrices, we adopt the spatial pyramid division mechanism discussed in the next subsection.

2.4. Spatial pyramid division

Given a collection of scale- t_k HKSI matrices $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$, we divide any element $\mathcal{S}_{t_k}^i$ into regular 4×4 grids at three levels (the reason for choosing the level number to 3 is to keep a balance between the experimental performance and the computational complexity), as illustrated in the upmost part of Fig. 4. For convenience, let $\mathcal{G}_{t_k}^{i\ell}$ denote any grid of $\mathcal{S}_{t_k}^i$ at level ℓ . Thus, we have the scale- t_k HKSI grid matrix collection $\{\mathcal{G}_{t_k}^{i\ell}\}_{i=1}^{\mathcal{I}}$ at level ℓ . In the presence of new frames during tracking, $\{\mathcal{S}_{t_k}^i\}_{i=1}^{\mathcal{I}}$ is extended along \mathcal{I} , as illustrated in the middle part of Fig. 4. Supposing there are K heat kernel diffusion scales, we have the multi-scale and multi-level HKSI grid matrix collections denoted as $\left\{ \left\{ \left\{ \mathcal{G}_{t_k}^{i\ell} \right\}_{i=1}^{\mathcal{I}} \right\}_{k=1}^K \right\}_{\ell=0}^2$. For each HKSI grid matrix collection $\mathcal{G}_{t_k}^{i\ell}$, an online multi-scale HKSI subspace model is learned,

as discussed in Sec. 2.3. It is noted that each HKSI grid matrix should be unfolded into a column vector before the step of subspace learning, as shown in the lower part of Fig. 4. Then, the incremental PCA algorithm [22] is used to learn a multi-scale HKSI subspace model at different levels in an online way. For a better illustration, we denote the multi-scale HKSI subspace model at level ℓ as $\mathcal{E}_{\mathcal{G}}^{\ell} = \{(U_{t_k\mathcal{G}}^{\ell}, \bar{\mathcal{S}}_{t_k\mathcal{G}}^{\ell})\}_{k=1}^K$ (i.e., the projection matrix and the mean HKSI matrix). According to Eq. (3), we have the likelihood evaluation mechanism for $\mathcal{E}_{\mathcal{G}}^{\ell}$ and multi-scale test HKSI grid matrices $\{\mathcal{T}_{t_k\mathcal{G}}^{\ell}\}_{k=1}^K$:

$$RE_{\mathcal{G}}^{\ell} = \frac{1}{K} \sum_{k=1}^K \|\text{UM}(\mathcal{T}_{t_k\mathcal{G}}^{\ell} - \bar{\mathcal{S}}_{t_k\mathcal{G}}^{\ell}) - U_{t_k\mathcal{G}}^{\ell} (U_{t_k\mathcal{G}}^{\ell})^T \text{UM}(\mathcal{T}_{t_k\mathcal{G}}^{\ell} - \bar{\mathcal{S}}_{t_k\mathcal{G}}^{\ell})\|_F^2 \quad (4)$$

Finally, the total likelihood evaluation score is defined as:

$$\mathcal{L}_{total} \propto \exp\left(-\frac{1}{3} \sum_{\ell} \sum_{\mathcal{G}} \omega_{\mathcal{G}}^{\ell} \cdot RE_{\mathcal{G}}^{\ell}\right) \quad (5)$$

where $\omega_{\mathcal{G}}^{\ell}$ is the weight associated with the HKSI grid \mathcal{G} at level ℓ (s.t. $\omega_{\mathcal{G}}^{\ell} = \frac{1}{4^{\ell}}$ in the experiments).

2.5. Bayesian tracking state inference

During visual tracking, a particle filter [2] is used for approximating the distribution over the object location using a set of weighted samples. The particle filter is governed by the observation model $p(O_t|X_t)$ and dynamic model $p(X_t|X_{t-1})$. Moreover, we apply an affine image warping to model the object motion of two consecutive frames. The six parameters of the affine transform are used to model $p(X_t|X_{t-1})$. Let $X_t = (x_t, y_t, \eta_t, s_t, \beta_t, \phi_t)$ where $x_t, y_t, \eta_t, s_t, \beta_t, \phi_t$ denote the x, y translations, the rotation angle, the scale, the aspect ratio, and the skew direction at time t , respectively. We employ a Gaussian distribution to model the state transition distribution $p(X_t|X_{t-1})$. Also the six parameters of the affine transform are assumed to be independent. Consequently, $p(X_t|X_{t-1})$ is formulated as: $p(X_t|X_{t-1}) = \mathcal{N}(X_t; X_{t-1}, \Sigma)$, where Σ denotes a diagonal covariance matrix with diagonal elements: $\sigma_x^2, \sigma_y^2, \sigma_{\eta}^2, \sigma_s^2, \sigma_{\beta}^2, \sigma_{\phi}^2$, respectively. The observation model $p(O_t|X_t)$ reflects the probability that a sample is generated from the HKSI subspaces. Consequently, $p(O_t|X_t)$ is formulated as: $p(O_t|X_t) \propto \mathcal{L}_{total}$, where \mathcal{L}_{total} is defined in Eq. (5). After maximum a posterior (MAP) estimation, we just use the multi-scale and multi-level HKSI grid matrices from the affinely warped image region associated with the highest weighted hypothesis to update the HKSI subspace models.



Figure 5: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 1 over representative frames (17, 27, 33, 39, 43, and 46) from Video 1 in the scenarios of image distortions and partial occlusions.



Figure 6: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 2 over representative frames (42, 44, 47, 55, 58, and 61) from Video 2 in the scenario of large image distortions.

3. Experiments

Eight experiments are conducted to demonstrate the advantages of our tracking framework based on HKSI subspace learning (referred to as *HKSL*). In these experiments, we compare the tracking results of *HKSL* with those of a representative incremental PCA based tracking method [22], referred to as *IPCA*, in different scenarios including pose variations, image distortions, scene blurring, occlusions, and small objects. *IPCA* is a representative subspace learning based tracking algorithm which uses the image-as-vector technique for object representation. By using online PCA, *IPCA* adapts to the undergoing object appearance changes, resulting in a presumably robust tracking result. In contrast to *IPCA*, *HKSL* relies on multi-scale HKSI subspace learning to reflect the appearance changes of an object. Consequently, it is interesting and desirable to make a comparison between *IPCA* and *HKSL*. The parameter settings of *IPCA* can be found in [22]. The following is a brief introduction to the experimental configurations of *HKSL*. In our multi-scale HKSI subspace models, the heat diffusion scaling factor t_k is chosen in a log-linear manner. To avoid the expensive computational costs due to intro-

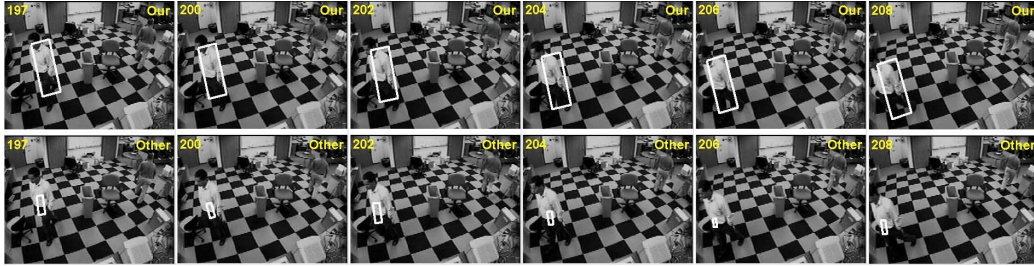


Figure 7: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 3 over representative frames (197, 200, 202, 204, 206, and 208) from Video 3 in the scenarios of image distortions and complex backgrounds.



Figure 8: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 4 over representative frames (90, 92, 117, 121, 146, and 149) from Video 4 in the scenario of occlusions.

ducing too many heat diffusion scales, we just consider three cases (i.e., $K = 3$) of t_k , i.e., $t_1 = 0.1$, $t_2 = 1$, and $t_3 = 10$. For image warping, the size of each object region is normalized to 32×32 pixels. The subspace dimension of $U_{t_k}^\ell \mathcal{G}$ in Eq. (4) is set as 16. For the particle filtering, the number of particles is set to 200. The six diagonal elements $(\sigma_x^2, \sigma_y^2, \sigma_\eta^2, \sigma_s^2, \sigma_\beta^2, \sigma_\phi^2)$ of the covariance matrix Σ in Sec. 2.5 are assigned as $(5^2, 5^2, 0.03^2, 0.03^2, 0.005^2, 0.001^2)$, respectively.

Next, we give a detailed description of the aforementioned eight experiments, including experimental purposes, video datasets, and tracking results. In the experiments, eight videos are used to make performance evaluations. All the eight videos consist of 8-bit gray scale images.

Experiment 1 is to compare the tracking performances of *IPCA* and *HKSL* using Video 1, where a pedestrian walks from left to right in a scene with an image distortion condition. His body pose varies over time, with a partial occlusion in the middle of the video stream. Samples of the final tracking results are demonstrated in Fig. 5.

Experiment 2 is for a tracking performance comparison between *IPCA* and *HKSL* using Video 2, in which a pedestrian moves in an indoor scene with large

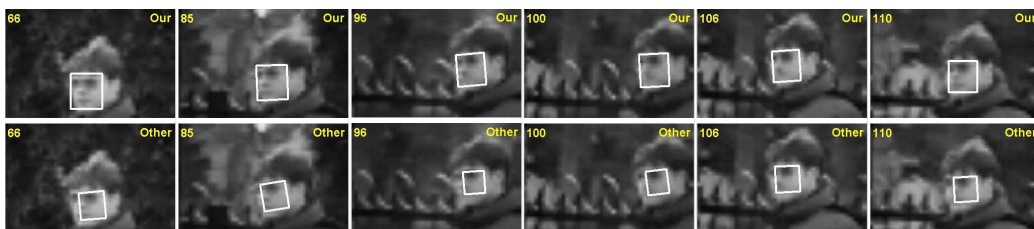


Figure 9: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 5 over representative frames (66, 85, 96, 100, 106, and 110) from Video 5 in the scenario of blurring.



Figure 10: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 6 over representative frames (18, 21, 29, 34, 51, and 60) from Video 6 in the scenarios of blurring and pose variations.

image distortions. Fig. 6 exhibits the tracking performances of *IPCA* and *HKSL*.

Experiment 3 aims to compare the tracking performance of *IPCA* with that of *HKSL* using Video 3, where a man moves in an indoor scene with image distortions and complex backgrounds. We show samples of the final tracking results for *HKSL* and *IPCA* in Fig. 7.

From the first three experiments, we see that *HKSL* is capable of tracking the object all the time in the cases of image distortions while *IPCA* fails. A brief theoretical analysis is given as follows. In the presence of image distortions, the structural properties (e.g. object shape) of object appearance vary. In this case, the image-as-vector *IPCA* performs badly because of ignoring the spatial interaction information of object appearance. In comparison, *HKSL* constructs an object representation using heat kernel structural information matrices on the object appearance graph. The object representation makes a full use of the interactive edge flow (or structural) information among the nodes of the object appearance graph, resulting in the insensitivity to image distortions.

Experiment 4 is to make a tracking performance comparison between *IPCA* and *HKSL* using Video 4, in which a man’s face is occluded by his hands from time to time. From Fig. 8, it is clear that *HKSL* is capable of tracking the object

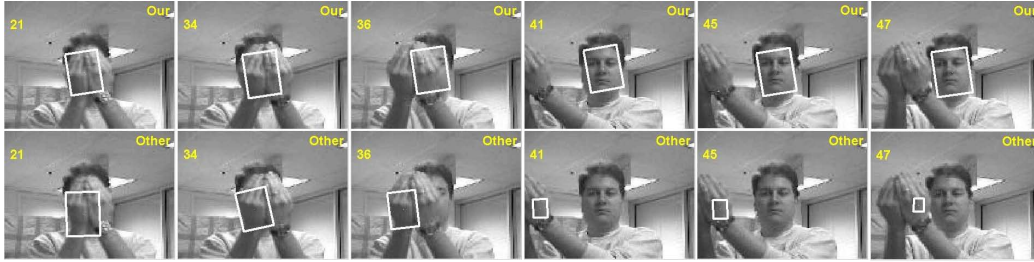


Figure 11: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 7 over representative frames (21, 34, 36, 41, 45, and 47) from Video 7 in the scenario of occlusions.

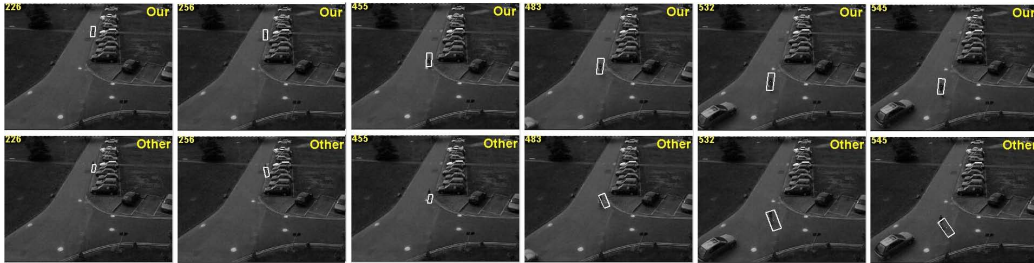


Figure 12: The tracking results of *HKSL* (row 1) and *IPCA* (row 2) in Experiment 8 over representative frames (226, 256, 455, 483, 532, and 545) from Video 8 in the scenarios of small object and blurring.

undergoing occlusions successfully while *IPCA* fails in tracking the object.

Experiment 5 aims to compare the tracking performance of *HKSL* with that of *IPCA* using Video 5, where a man with pose variations moves from right to left in a blurring outdoor scene. Fig. 9 shows the tracking performances of *HKSL* and *IPCA* in . Clearly, *HKSL* tracks the object in the blurring scenario more accurately than *IPCA*.

Experiment 6 is for a comparison between *IPCA* and *HKSL* on pose variations and scene blurring using Video 6, which is composed of quite low-quality images, as a number of vehicles moves in a highway. From Fig. 10, *HKSL* tracks the object all the time while *IPCA* fails to track the object soon after a few frames.

Experiment 7 is to make a performance comparison between *IPCA* and *HKSL* using Video 7, in which a man's face is occluded by his hands from time to time. Fig. 11 demonstrates the final tracking results. From Fig. 11, it is clear that *HKSL* is capable of tracking the object successfully while *IPCA* fails to track the object. The reason is that the multi-level spatial pyramid HKSI grid subspace information is incorporated into *HKSL*. Even if the subspace information of some spatial pyramid HKSI grids is lost or drastically varies, *HKSL* can still utilize the remaining

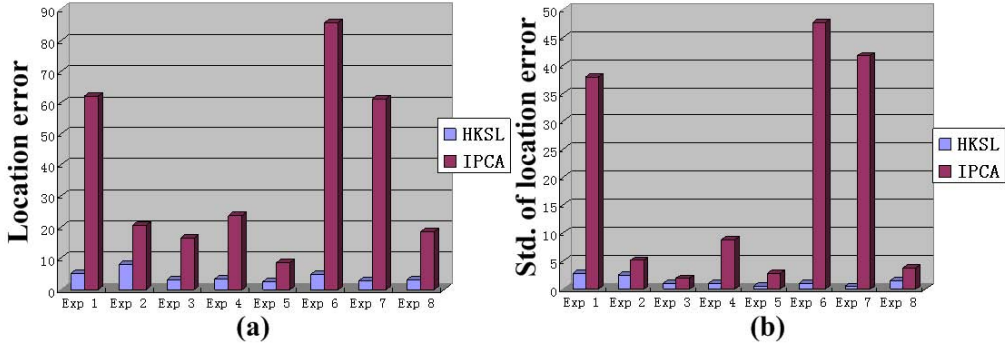


Figure 13: **The quantitative comparison results of *HKSL* and *IPCA*. Exp k corresponds to Experiment k for $1 \leq k \leq 8$. (a) shows the mean tracking location errors of the eight experiments, and (b) displays the standard deviations of the frame-dependent tracking location errors of the eight experiments.**

subspace information from other spatial pyramid HKSI grids to accomplish the tracking task.

The last experiment is to investigate the tracking performances of *IPCA* and *HKSL* in handling scene blurring and small object scenarios using Video 8, in which a pedestrian as a small object moves down a road in a dark and blurry scene. Samples of the final tracking results for *HKSL* and *IPCA* are shown in Fig. 12. Clearly, *HKSL* succeeds in tracking the pedestrian while *IPCA* fails.

In order to make a quantitative comparison between *HKSL* and *IPCA*, the object center locations are labeled manually as the ground truth. In this case, we can quantitatively evaluate the tracking performances of *HKSL* and *IPCA* by computing their corresponding pixel-based tracking location errors with the ground truth. The final comparing results are reported in Fig. 13. More specifically, Fig. 13(a) shows the mean tracking location errors of the eight experiments, and Fig. 13(b) displays the standard deviations of the frame-dependent tracking location errors of the eight experiments. The smaller the mean tracking location error, the more accurate the tracker. The smaller the standard deviation, the more stable the tracker. From Fig. 13(a), we can see that the mean location errors and the standard deviations of *HKSL* in the eight experiments are always lower than those of *IPCA*. Thus, *HKSL* achieves the more accurate and more stable tracking performances than *IPCA*.

In summary, we observe that *HKSL* outperforms *IPCA* in the scenarios of blurring, pose variations, image distortions, occlusions, and small objects. *HKSL* makes a full use of the multi-scale spatial structural information of object ap-

pearance. In comparison, *IPCA* only captures the statistical properties of object appearance in an image-as-vector manner, resulting in the loss of the local spatial structural information in the object region. Consequently, *HKSL* provides a better way of modeling appearance changes of an object in many complex scenarios.

4. Conclusion

In this paper, we have proposed a tracking framework based on multi-scale heat kernel structural information (HKSI) matrices. In this framework, an object is represented as multi-scale HKSI matrices which perform well in characterizing the edge flow (or structural) information on the object appearance graph. Further, we have taken a spatial pyramid division strategy for capturing the multi-level spatial layout information of HKSI matrices. The incremental PCA algorithm has been used to learn online grid-specific HKSI subspace models reflecting the appearance change of an object. Then, the grid-specific HKSI subspace models serve as the observation model of a particle filter for tracking. Moreover, a novel criterion for the likelihood evaluation, based on multi-scale HKSI subspace reconstruction error norms, has been proposed to measure the similarity between the test image and the learned HKSI subspace models during tracking. Experimental results have demonstrated that our framework achieves better tracking performances in the scenarios of blurring, pose variations, image distortions, occlusions, and small objects.

References

- [1] G. Hager and P. Belhumeur, "Real-time Tracking of Image Regions with Changes in Geometry and Illumination," in *Proc. CVPR*, pp.430-410, 1996.
- [2] M. Isard and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," in *Proc. ECCV*, Vol. 2, pp.343-356, 1996.
- [3] Q. Zhao, S. Brennan, and H. Tao, "Differential EMD Tracking," in *Proc. ICCV*, 2007.
- [4] G. Silveira and E. Malis, "Real-time Visual Tracking under Arbitrary Illumination Changes," in *Proc. CVPR*, 2007.
- [5] M. J. Black, D. J. Fleet, and Y. Yacoob, "A Framework for Modeling Appearance Change in Image Sequence," in *Proc. ICCV*, pp.660-667, 1998.
- [6] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust Online Appearance Models for Visual Tracking," in *Proc. CVPR*, Vol. 1, pp.415-422, 2001.
- [7] S. K. Zhou, R. Chellappa, and B. Moghaddam, "Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters," *IEEE Trans. on Image Processing*, Vol. 13, pp.1491-1506, November 2004.

- [8] H. Wang, D. Suter, K. Schindler, and C. Shen, "Adaptive Object Tracking Based on an Effective Appearance Filter," *IEEE Trans. on PAMI.*, Vol. 29, Iss. 9, pp.1661-1667, 2007.
- [9] A. Yilmaz, "Object Tracking by Asymmetric Kernel Mean Shift with Automatic Scale and Orientation Selection," in *Proc. CVPR*, 2007.
- [10] Q. A. Nguyen, A. Robles-Kelly, and C. Shen, "Kernel-based Tracking from a Probabilistic Viewpoint," in *Proc. CVPR*, 2007.
- [11] T. Yu and Y. Wu, "Differential Tracking based on Spatial-Appearance Model(SAM)," in *Proc. CVPR*, Vol. 1, pp.720-727, June 2006.
- [12] J. Li, S. K. Zhou, and R. Chellappa, "Appearance Modeling under Geometric Context," in *Proc. ICCV*, Vol. 2, pp.1252-1259, 2005.
- [13] S. Ilić and P. Fua, "Non-Linear Beam Model for Tracking Large Deformations," in *Proc. ICCV*, 2007.
- [14] S. Tran and L. Davis, "Robust Object Tracking with Regional Affine Invariant Features," in *Proc. ICCV*, 2007.
- [15] M. Grabner, H. Grabner, and H. Bischof, "Learning Features for Tracking," in *Proc. CVPR*, 2007.
- [16] K. Lee and D. Kriegman, "Online Learning of Probabilistic Appearance Manifolds for Video-based Recognition and Tracking," in *Proc. CVPR*, Vol. 1, pp.852-859, 2005.
- [17] H. Lim, V. I. Morariu³, O. I. Camps, and M. Sznai¹, "Dynamic Appearance Modeling for Human Tracking," in *Proc. CVPR*, Vol. 1, pp.751-757, 2006.
- [18] M. J. Black and A. D. Jepson, "Eigenttracking: Robust matching and tracking of articulated objects using view-based representation," in *Proc. ECCV*, pp.329-342, 1996.
- [19] J. Ho, K. Lee, M. Yang and D. Kriegman, "Visual Tracking Using Learned Linear Subspaces," in *Proc. CVPR*, Vol. 1, pp.782-789, 2004.
- [20] Y. Li, L. Xu, J. Morphet^t and R. Jacobs, "On Incremental and Robust Subspace Learning," *Pattern Recognition*, 37(7), pp. 1509-1518, 2004.
- [21] D. Skocaj and A. Leonardis, "Weighted and Robust Incremental Method for Subspace Learning," in *Proc. ICCV*, pp.1494-1501, 2003.
- [22] D. Ross, J. Lim, R. Lin, and M. Yang. "Incremental Learning for Robust Visual Tracking," *IJCV*, 77(1-3):125-141, 2008.
- [23] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo, "Robust Visual Tracking Based on Incremental Tensor Subspace Learning," in *Proc. ICCV*, 2007.
- [24] F. Porikli, O. Tuzel, and P. Meer, "Covariance Tracking using Model Update Based on Lie Algebra," in *Proc. CVPR*, Vol. 1, pp. 728-735, 2006.
- [25] X. Li, W. Hu, Z. Zhang, X. Zhang, M. Zhu, and J. Cheng, "Visual Tracking Via Incremental Log-Euclidean Riemannian Subspace Learning," in *Proc. CVPR*, 2008.
- [26] X. Li, W. Hu, Z. Zhang, and X. Zhang, "Robust Visual Tracking Based on An Effective Appearance Model," in *Proc. ECCV*, 2008.
- [27] X. Bai and E. R. Hancock, "Heat Kernels, Manifolds and Graph Embedding," in *Structural, Syntactic, and Statistical Pattern Recognition*, pp. 198-206, 2004.

- [28] B. Xiao, R. C. Wilson, and E. R. Hancock, “Characterising Graphs using the Heat Kernel,” in *Proc. BMVC*, 2005.
- [29] Y. Wu, J. Cheng, J. Wang, and H. Lu, “Real-time Visual Tracking via Incremental Covariance Tensor Learning,” in *Proc. ICCV*, 2009.
- [30] B. Babenko, M. Yang, and S. Belongie, “Visual Tracking with Online Multiple Instance Learning,” in *Proc. CVPR*, 2009.