EIGEN-PROFILES OF SPATIO-TEMPORAL FRAGMENTS FOR ADAPTIVE REGION-BASED TRACKING

Adway Mitra[1]*, Anoop K.R.*[2]*, Ujwal Bonde*[2]*, Chiranjib Bhattacharyya*[1]*, K.R.Ramakrishnan*[2]

[1]Department of Computer Science, [2]Department of Electrical Engineering
Indian Institute of Science,Bengaluru,India
{{anoopkr,krr}@ee,{adway,chiru}@csa}.iisc.ernet.in

ABSTRACT

We propose a novel space-time descriptor for region-based tracking which is very concise and efficient. The regions represented by covariance matrices within a temporal fragment, are used to estimate this space-time descriptor which we call the *Eigenprofiles*(EP). EP so obtained is used in estimating the Covariance Matrix of features over *spatio-temporal fragments*. The Second Order Statistics of spatio-temporal fragments form our target model which can be adapted for variations across the video. The model being concise also allows the use of multiple spatially overlapping fragments to represent the target. We demonstrate good tracking results on very challenging datasets, shot under insufficient illumination conditions.

Index Terms— Covariance Matrix, Eigenvectors, Tracking, Joint Diagonalization, Spatio-Temporal Fragment

1. INTRODUCTION

Research in tracking has aimed to build efficient appearance models for objects that are robust to real-world challenges, like illumination changes, variations in pose, scale, shape etc. Efficient and concise space-time respresentation of objects being tracked is thus a challenging task. In tracking literature, three main approaches have been proposed for target modeling. They are low-level (pixel-based), mid-level (region-based) and high-level (parts, shape or pose based). Low-level approaches include tracking interest points such as SIFT [1] and high-level approaches involve building more sophisticated models, like the individual body parts of a human, and tracking then simultaneously [2]. But in videos shot under insufficient illumination, the individual parts are often not visible and the frames are noisy and grainy, rendering interest Points very unreliable. Rather, the target appears like a blob or patch, which suggests a region-based (mid-level) approach. Region-based tracking requires efficient region descriptors,like Colour Histograms [3]. But a more powerful and efficient method is the Covariance Descriptor [4], which is used for Region tracking [5]. Covariance Descriptor of a region in an image is the sample covariance matrix of the feature vectors at locations within that particular region. This descriptor has been shown to be robust to noise and scale.

In this paper, we propose a novel space-time descriptor which we call the Eigenprofile. Estimation of EP is equivalent to joint diagonalization of these covariance matrices and they form a matrix of orthonormal vectors. We incrementally build models for the target using EP, making use of the property that the appearance is more or less constant over short sliding time windows. We use the term **spatial fragment**(SF) to indicate a region within an image, and **temporal fragment**(TF) as a collection of frames within a time-window.The cube formed by stacking the corresponding Spatial-Fragments, which are the spatial regions containing object within a frame, within a Time-Fragment as the **Spatio-Temporal-Fragment (STF)** . The second-order statistics of these STFs form our target model. The tracking proceeds by continuously adapting STF models from target SFs over sliding TFs, and matching candidate SFs in new frames to these models.

The original Covariance Tracker [4] also models the target by a statistic of target SFs over a TF. This statistic is the *Intrinsic Mean* of the SF Covariance Matrices from these frames. Wu et al propose [6] for learning another statistic for STFs, which is equivalent to pooling the features from target SFs in different frames together and estimating the Covariance Matrix. In tracking literature, a recent paradigm is the fragment-based approach [7], where multiple image patches (SFs) are used to build a template for an object within a single frame. Increasing the number of SFs improve the tracking performance, but also require larger model size. Our approach provides considerable efficiency in storage, and this efficiency can be utilized to increase the number of SFs.

2. EIGENPROFILES

Consider a TF of K frames, where we have Covariance Matrices $C_{t+1}, C_{t+2}, \dots, C_{t+K}$ for corresponding SFs. In Video-based applications, we observe empirically that the p-dimensional feature vectors in corresponding SFs in the individual frames within a TF have *almost identical principal components*, which are nothing but the eigenvectors of

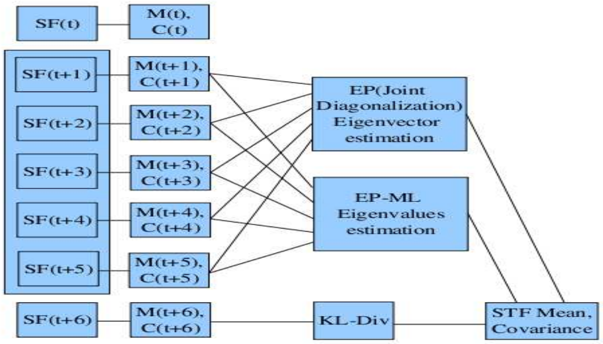

Fig. 1. schematic diagram of the tracking. We use SF models from a TF of 5 successive frames to be build STF model, and compare candidate SFs from the next frame with it.

the corresponding covariance matrices, ordered with respect to the eigenvalues. Hence, we propose to approximate the eigenbases of the $\{C_k\}$ matrices with a common eigenbasis which we call the $Eigenprofile$ of that STF.

2.1. Estimation of the Eigenprofile

Each SF covariance matrix within a TF can be expressed completely with its eigenvectors and eigenvalues as $C_k = \sum_j \delta_{kj} e_{kj} {e_{kj}}^T$. Under our hypothesis, it can be approximated by shared eigenvectors as

$$C_k \approx \sum_{j=1}^{p} \delta_{kj} \beta_j {\beta_j}^T \tag{1}$$

Here the β vectors form the EP for the STF obtained by stacking these K SFs. Estimation of EP is nothing but *Approximate Joint Diagonalization* of the $\{C_k\}$ matrices. There is a family of Approximate Joint Diagonalization algorithms, of which one is by Pham [8]. Given the C_k matrices, this algorithm attempts to find a single matrix V to minimize the following function

$$\sum_k (log(det(diag(V^T C_k V))) - log(det(V^T C_k V))) \tag{2}$$

However these algorithms do not make use of similarity of eigenvectors of the input matrices in any way. Here, we propose to use this additional information to make an improved estimate. We formulate the optimization problem as

$$\min_{\beta} \sum_{k=t+1}^{t+K} \|C_k - \sum_{j=1}^{p} \delta_{kj} \beta_j \beta_j^{\mathsf{T}} \|_F^2 \text{ such that} \tag{3}$$

$$\beta_j^T \beta_j = 1 \forall j \text{ and } \beta_j^T \beta_i = 0 \forall i \neq j \tag{4}$$

Writing the Lagrangian dual and solving it with respect to β, we have

$${\beta_j}^T D_j \beta_j = \alpha_j \tag{5}$$

where $D_j = \sum_{k=t+1}^{t+K} 2\psi_{kj} C_k$, which is a symmetric matrix. The program is not convex, but a local solution is obtained when we have β_j as an eigenvector of D_j, for every j. Then we require the D_j matrices for estimating the Eigenprofile. But, we would like to have an estimate of EP from the eigenvectors of the $\{C_k\}$ matrices directly so that we do not need to store the entire matrices. To facilitate this, we use the observation that the corresponding eigenvectors of the $\{C_k\}$ matrices are quite identical to each other, i.e. $e_{(t+1)j} \approx e_{(t+2)j} \approx \cdots \approx e_{(t+K)j}$

Hence, we solve the following optimization problem

$$\min_{u_j^T u_j = 1} \sum_{k=t+1}^{t+K} ||u_j - e_{kj}||^2 \quad \text{subject to}$$

The solution to this is an estimate of the i-th eigenvector of D_j, and is given by

$$u_j = \frac{\sum_k e_{kj}}{\sqrt{(\sum_k e_{kj})^T (\sum_k e_{kj})}} \tag{6}$$

It is to be noted that the estimates u_j of β_j thus obtained do not satisfy the orthogonality criteria, as required by the definition of Eigenprofile. So, we orthonormalize them by Gram-Schmidt procedure, to obtain orthonormal $\{\beta_j\}$.

3. ESTIMATION OF STF COVARIANCE MATRIX

For the tracking application, we build the Covariance Matrix C of the STF as the target model. We posit that C will have the Eigenprofile β as eigenvectors. Hence C is given by $C = \sum_j \sigma_j \beta_j {\beta_j}^T$. So we are now left with the estimation of its *eigenvalues* σ_j to learn it completely.

3.1. Maximum Likelihood Estimate: EP-ML

We estimate the STF Covariance Matrix C using Maximum-Likelihood Estimate (**EP-ML**). Within a temporal fragment, the feature vectors in corresponding spatial fragments of individual frames should follow the same distribution. It is known that sample Covariance Matrices of sample populations drawn from the same distribution follow the Wishart Distribution. Assuming these sample covariance matrices C_k are I.I.D., the probability of this set is given by

$$p(\{C_k\}|C) = T \frac{\prod_k |C_k|^{\frac{-p}{2}} e^{(-\frac{1}{2}(trace(C^{-1} \sum_k C_k))}}{|C|^{\frac{K}{2}}} \tag{7}$$

By differentiating with respect to σ_j and equating to 0, the M.L.E. of the eigenvalues σ_j from Equation 7 is given by

$$\sigma_j = \frac{\sum_k {\beta_j}^T C_k \beta_j}{K} \approx \frac{\sum_k \delta_{kj}}{K} \tag{8}$$

3.2. Low-Rank Approximation of STF Covariance Matrix

For many features, including GaborFeatures which we have used in our experiments,it is observed that the leading eigenvalues of Covariance Matrices of the SFs are considerably larger compared to the rest, which rapidly trail off towards zero. Equation 8 shows that the same has to hold for the eigenvalues of the STF Covariance Matrix, and so it is possible to approximate the STF Covariance Matrix with only its R leading eigenpairs, as $C_{low} = \sum_{1 \leq j \leq R} \sigma_j \beta_j {\beta_j}^T$. Thus, for p-dimensional features, STF model now consists of the STF Mean Vector μ, R EP-vectors β of dimension p, and R eigenvalues σ. Moreover we need not store the ST matrices C_k from the frames, but only the R leading eigenvalues δ_k, the corresponding eigenvectors e_k and the mean vector μ_k of the SF. The mean vector μ for STF can be easily obtained from the SF mean vectors μ_k in the individual frames of the TF, as $\mu = \dfrac{\sum_k n_k \mu_k}{\sum_k n_k}$, n_k being number of feature vectors in the SF in k-th frame. Thus such an approximation of the matrix results in some storage efficiency, especially when R is considerably lower than p.

4. TRACKING

We now proceed to describe the framework of tracking we used in the experiments. As the main aim of the paper is to build a model and not a tracker, we restrict ourself to a simple but effective tracking framework.

4.1. Spatio-Temporal Fragments

As mentioned earlier, in our tracking experiments we use multiple spatially overlapping fragments to model the target. We build **9 STF models**. If in a particular frame the object is known to be located inside a tight rectangle centered at (x, y) with length and breadth (dx, dy), the mean vector and SF Covariance Matrix of features from this rectangular SF are used to build the **Central Model**, and 8 **Peripheral Models** are obtained from the Mean Vectors and SF Covariance Matrices of the rectangular SFs centered at $(x + \delta_x dx/2, y + \delta_y dy/2)$ with dimensions (dx, dy), where $\delta_x, \delta_y \in \{1, 0, -1\}$.

4.2. Dissimilarity Between Region Models

During tracking, given any new frame, we need to compare the SFs at the candidate locations against the target model(s), and report the location where the matching is the best. This requires a measure to compare the STF model(s) to the candidate SF model(s). In case of our EP-based method, a STF model consists of STF Mean Vector and STF Covariance Matrix. We use the KL-Divergence as the measure of dissimilarity. In case of Covariance Tracker [5] and ICTL [6], the measure is the **Geodesic Distance (GD)** between Covariance Matrices. We also implemented ICTL using KL-Divergence as this measure. We call this as **ICTL2** in our results. For Pham's Algorithm of Joint Diagonalization [8], the measure in Equation 2 is used. In this case, the STFs are represented by the V matrix of 2, output by Pham's algorithm. Since we have 9 STF models $R_1, R_2, \ldots, R_9$ as mentioned above, at each candidate location (x, y) we get 9 candidate SF models $C_1, C_2, \ldots, C_9$. We compare the candidate models to the corresponding STF models to get a final score $f(x, y) = \sum_{i=1}^{9} KL(R_i, C_i)$. In cases where GD or Equation 2 is used, the function f is modified suitably.

The algorithm is described in details in the adjacent box.

Algorithm 1 Tracking Algorithm

```
Initialize the locations $X_1, X_2, X_3, X_4, X_5$ of the target and its size $(\delta_1, \delta_2)$ in the first 5 frames.
Crop out 9 rectangular SFs around $X_i$ and calculate their mean and SF Matrices for $1 \leq i \leq 5$.
Estimate the 9 STF models and save them.
for $i = FirstFrame : LastFrame$
    choose $N$ candidate locations
    for $i = 1 : N$
        Crop out 9 rectangular SFs corresponding to central and peripheral models around candidate location $X_i$
        Build the candidate SF models $C_1, C_2, \ldots, C_9$ from these.

        Calculate $f(X_i)$ with the respective STF models $R_1, R_2, \ldots, R_9$
    end for
    Set the location to $(X^*)$ where $f(X^*)$ is minimum among all candidate locations
    Re-estimate the 9 STF models by replacing the oldest frame in the TF with the current one
end for
```

5. EXPERIMENTAL EVALUATION

5.1. Datasets and Features

We have carried out experiments on 9 datasets. Of these 2 are standard and 7 captured by us. We have used one sequence (SEQ1) from PETS2000. SEQ2 is the publicly available Toni dataset ([9]) which involves tracking the face of a person in an obscure room. The person also turns his head and there is a sudden illumination change. SEQ3 and SEQ4 are indoor videos of a person walking on a long corridor with a single light. In SEQ3 the person walks into an obscure area with sharp illumination gradient and in SEQ4 initially the light is off, then it comes on and finally goes off again. The background also changes considerably. SEQ5-SEQ9 are all outdoor videos captured at night. In all the cases there is minimal lighting, and it is difficult to distinguish the target from the background clearly. In SEQ6, in the beginning the person

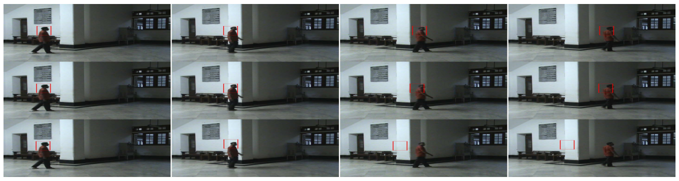

Fig. 2. SEQ3: The results shown are for EP-ML,ICTL and Covariance Tracker from top to bottom. COV losse track at the illumination gradient in the middle

SEQ	EP-ML	IVT	COV	ICTL	ICTL2	Pham
1	**0**	**0**	0.37	0.02	**0**	0.63
2	**0.07**	0.68	0.70	0.70	0.16	0.38
3	**0**	0.50	0.50	**0**	**0**	0.50
4	**0**	0.72	0.70	**0**	**0**	0.70
5	**0**	0.65	**0**	**0**	**0**	0.08
6	**0**	0.42	0.93	0.93	0.93	0.93
7	**0**	**0**	0.67	0.72	**0**	**0**
8	**0**	0.24	0.41	0.49	0.36	0.33
9	**0**	0.72	0.86	0.50	0.50	0.78

Table 1. Fraction of frames in the videos where the output's overlap with Ground Truth is 0 pixels.

moves before an unevenly lit background. Moreover, there is a sudden illumination change of the target due to the lights of a passing vehicle. The videos are of varying length with as short as 39 frames (SEQ1) to as long as 600 (SEQ2). We used 12-dimensional Gabor Features (3scales, 4orientations) for the sequences. In the eigenprofile-based method, only the leading 3 eigenvectors are used for low-rank approximation of the STF Covariance Matrix.

5.2. Benchmark Methods and Results

Since the proposed approach is region-based, and uses Covariance Matrices to model regions, we compare with related approaches like Covariance Tracker and ICTL. Again, as EP is obtained by Joint Diagonalization, we compare with an alternative JD algorithm ([8]). **All these experiments were performed under the same basic framework of features and Tracking Algorithm, with only the STF model differing across the methods.** Moreover, we also show results using IVT as in [10] which is not covariance-based, but well-known. In the Ground Truth, the target's locations are specified by a tight rectangle around it. During tracking also, the method marks the inferred region with a rectangle. The number of frames in which the overlap of these two rectangles is 0 is provided in Table 1. It can be seen that our method EP-ML achieve the best performance in all the 9 sequences. The code and data are available in $http : //clweb.csa.iisc.ernet.in/adway/tracking/$.

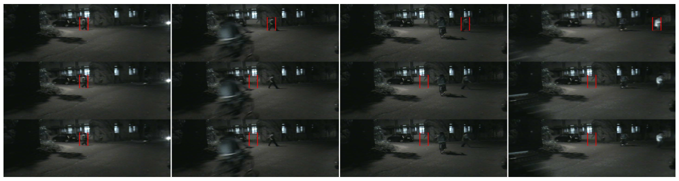

Fig. 3. SEQ6: EP-ML,ICTL and Cov. Tracker from top to bottom. The video is dark and blurred, EP-ML succeeds unlike the rest

6. CONCLUSION

The experiments clearly show the superior performance of the EP-based method over other approaches, under the same framework. Moreover, the EP-estimation is also computationally far more efficient than the Covariance Tracker, which involves an iterative algorithm at each frame to calculate the Intrinsic Mean Matrix. Also, our method provides the additional advantage of storage-efficiency. This efficiency can be utilized in increasing the number of spatial fragments without increasing the memory footprint. The framework currently does not have any motion-model, inclusion of which can turn it into a highly accurate tracker.

7. REFERENCES

[1] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. Freeman, "Sift flow: dense correspondence across different scenes," *Computer Vision–ECCV 2008*, pp. 28–42, 2008.

[2] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *IEEE CVPR, 2008*. IEEE, 2008, pp. 1–8.

[3] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE TPAMI*, pp. 564–575, 2003.

[4] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," *ECCV 2006*, pp. 589–600, 2006.

[5] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in *IEEE CVPR, 2006*. IEEE, 2006, pp. 728–735.

[6] Y. Wu, J. Cheng, J. Wang, and H. Lu, "Real-time visual tracking via incremental covariance tensor learning," in *IEEE ICCV, 2009*. IEEE, 2010, pp. 1631–1638.

[7] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *IEEE CVPR, 2006*. IEEE, 2006, pp. 798–805.

[8] D. T. Pham, "Joint approximate diagonalization of positive definite hermitian matrices," *SIAM J. Matrix Anal. Appl.*, vol. 22, no. 4, 2001.

[9] "http://www.svcl.ucsd.edu/projects/tracking/results.html," .

[10] D.A. Ross, J. Lim, R.S. Lin, and M.H. Yang, "Incremental learning for robust visual tracking," *IJCV*, vol. 77, no. 1, pp. 125–141, 2008.