A Covariance Approximation on Euclidean Space for Visual Tracking

Stefan Kluckner, Thomas Mauthner, and Horst Bischof

Institute for Computer Graphics and Vision
Graz University of Technology, Austria
$\{$*kluckner,mauthner,bischof*$\}$@*icg.tugraz.at*

Abstract
This work proposes an efficient approximation of a covariance based feature representation for tracking. In contrast to approximated similarity measurements between second order moments, such as the Foerstner metric, we propose to approximate single distributions by specified sampling. We derive an efficient and discriminative feature representation that allows to compute distances between covariance-based descriptors on Euclidean space. This approximated representation fits perfectly to the application of tracking, where efficient similarity measurement significantly controls the efficiency and the real-time capability of the resulting approach. Furthermore, we highlight the advantages of the proposed approximation for learning an object-specific representation during tracking. The experimental evaluation shows results on standard tracking videos and compares our derived approach to state-of-the-art methods based on other covariance representations.

1 Introduction

Object tracking is an important task in many computer vision applications such as visual surveillance, human computer interaction, traffic monitoring, augmented reality and sports analysis. The handling of pose and appearance variations of the tracked object is a fundamental and challenging task. In addition, many extrinsic influences such as cluttered background, multiple objects, variations in illumination and moving cameras complicate the task. Robust visual tracking therefore depends on discriminative appearance models and robust and efficient updates during tracking. A variety of tracking algorithms have been proposed to overcome these difficulties.

Many efficient tracking methods are based on color histograms to describe the appearance of the tracking target. Comaniciu et al. [5] proposed the mean-shift method, a non-parametric density gradient estimator, where the image window most similar to the previous object representation, is found iteratively by carrying out a kernel based search. Under large frame-to-frame displacements, where the object kernel has no overlap with the previous position of the object, the mean-shift tracker tends to get stuck in local minimas. To overcome this problem, probabilistic methods use the objects state space to model the underlying dynamics of the tracking system. One popular approach is the particle filter [16], also known as sequential *Monte Carlo* methods [1]. Particle filters can be interpreted as a probabilistic search algorithm, where a set of particles, each describing one possible state, models the posterior probability representing the current knowledge about the overall object state. In the computer vision community particle filtering techniques have been widely applied to tracking problems where it is also referred to as *Condensation* algorithm [8]. Maggio and Cavallaro [15] combined particle filters and the mean-shift approach into a two stage tracker, where the particles are shifted toward the nearest maximum after the resampling step. Nevertheless, using only color histograms limits the discrimantion power, which influences the tracking results significantly. Birchfield and Rangarajan [4] proposed the concept of spatiograms, where spatial statistical information is given for each bin of a generated histogram. Experiments showed improved tracking results compared to standard histogram approaches. Wang et al. [24] introduced an adaptive appearance model based on joint spatial-color Gaussian mixture models. To avoid the dependency on single feature cues, the integration of several independent features increases the capability to react on critical tracking situations. Maggio et al. [15] calculated several uncertainties for particle weights from different feature cues, color and edge histograms, and used their proportions to dynamically adjust their influence onto the tracking result in an on-line formulation. Recently, Badrinarayanan et al. [3] used a similar approach to estimate the uncertainty for a color based particle filter tracker. Their multi-cue tracking approach combines a novel randomized template tracker with a constant color model based particle filter by switching and interacting between the different feature cues. A more direct way to incorporate several feature cues for compact region based representation has been proposed by Tuzel et al. [20, 21]. Their proposed covariance descriptor captures spatial and statistical as well as correlation relation between features, while the dimensionality of the descriptor is kept small.

Furthermore, the capability to potentially adapt the object representation during the tracking process, possibly against the actual background, is of vital interest for robust tracking [7, 2, 14]. Avidan [2] considered tracking as a binary classification problem on pixel level. An ensemble of weak classifier is trained on-line to distinguish between the object and the current background, while a subsequent mean-shift procedure obtains the exact object localization. Grabner et al. [7] proposed on-line AdaBoost for feature selection, where the object representation is trained on-line with respect to the current background. In contrast to [2], their classification was applied to image patches instead of pixels, combining various region-based feature descriptor for appearance and texture. Lim et al. [14] introduced an efficient on-line algorithm for incrementally learning the eigenspace representation of the tracked object, which facilitates the tracking task.

Our framework incorporates powerful covariance features for discriminative object representation and the capability to efficiently update the object representations during tracking directly on Euclidean space. Based on the approximated covariance representation describing both, texture and appearance, we propose a probabilistic tracking system, where the tracked object is represented using five sub-parts for additional robustness. Each sub-part is individually updated and evaluated using incremental PCA for a eigenspace representation. The main contribution of our work is twofold. First, we present the approximation for single covariance matrices using an efficient non-linear transform. By applying a *Cholesky* factorization, a feature representation is obtained, which is defined on Euclidean vector space and captures both, covariance and mean information. In contrast to typical operations on Riemannian manifolds the resulting feature representation can be directly used in standard update strategies and machine learning techniques. These computational advantages constitute the second contribution.

The remainder of this paper is structured as follows: In Section 2 we discuss the related work to covariance-based tracking. Section 3 briefly reviews the first and second order moments such as the mean and the covariance representation, the efficient computation using integral structures and describes the idea of the covariance approximation in detail. Section 4 highlights the application to tracking, reviews the part based particle filtering, and discusses the subspace learning for appearance modeling. In Section 5 experimental evaluations are given. Section 6 concludes our work and gives an outlook on future work.

2 Related Work on Covariance Based Tracking

As proposed in [17], covariance descriptors for tracking and detection can be computed efficiently on Cartesian space using an extension of single integral structures [23]. Due to the fact that covariance matrices do not lie on Euclidean feature space, non-linear mappings to Riemannian manifolds [20] or Lie algebra [17, 21] are used to obtain vector spaces in which the metrics for machine learning techniques are defined. In [17] the *Foerstner* metric [6] is applied to approximated covariance similarity measurements on the Riemannian manifold. Recently, in [21] Tuzel et al. treat tracking as a learning and detection problem, where the learning strategy estimates the affine motion using Lie algebra. They adopted a regression model for learning and modeled the appearance using orientation histograms. Yao and Odobez [25] proposed an extension of [20] for human tracking by detection, where they use feature selection by extracting sub-parts of covariance matrices to speed up the computation. Additionally, this approach incorporates the mean within the rectangular regions for a rejection step in the feature selection procedure. Li et al. [13] developed a tracking framework applying a $\log$-Euclidean Riemannian metric for on-line learning. They suggest to use a low-dimensional eigenspace for on-line updates over time and propose a probabilistic formulation of likelihood function based on the reconstruction error of the $\log$-Euclidean eigenspace model. Contrary to [25, 13, 21], Tyagi and Davis [22] applied an on-line filtering technique similar to *Kalman* filters [12] to model linear dynamical systems on Riemannian manifolds.

3 Approximated Mean and Covariance Representation

In this section we highlight the extraction of our compact region-based features derived from first and second order moments. As proposed in [19] discriminant covariance representations efficiently combine raw pixel values such as appearance and texture descriptions and give feasible results in tracking by detection applications [17, 20]. In contrast to these approaches, that are based on computationally expensive processing steps on Riemannian manifolds, our idea relies on approximating the mean and covariance representation on Euclidean vector space. In the following, we discuss the well studied mean and covariance descriptors, the computation and the idea of our approximation.

3.1 Mean and Covariance Descriptors

Computing the covariance region descriptor from multiple information sources yields a straightforward technique for a low-dimensional feature representation. A covariance matrix contains the variance of each source channel in its diagonal elements and the off diagonal elements describe the correlation values between the involved modalities. Considering an image I with the dimension $w \times h \times d$ any extracted covariance descriptor of an arbitrary rectangular size $N \times M$ results in a second order sample matrix $\mathbf{\Sigma}_r \in \mathbf{R}^{dxd}$:

$$\mathbf{\Sigma}_r = \frac{1}{NM-1} \sum_{x=1}^{N} \sum_{y=1}^{M} \left(I\left(x,y\right) - \mu \right) \left(I\left(x,y\right) - \mu \right)^T , \tag{1}$$

where $\mu \in \mathbf{R}^d$ is the sample mean vector:

$$\mu = \frac{1}{NM} \sum_{x=1}^{N} \sum_{y=1}^{M} I\left(x, y\right). \tag{2}$$

The feature vector $I\left(x, y\right)$ corresponds to a set of extracted values at the specified position x and y in the image I. These vectors are not restricted to normalized value ranges and include usually appearance, texture or spatial attributes such as color, derivatives, coordinates, etc. Extracting the covariance of an inhomogeneous area results in a strictly symmetric and positive semi-definite matrix with constant dimensions that models the properties of the specified region. A restriction to non-spatial attributes preserves the scale and rotation invariance because $\mathbf{\Sigma}_r$ does not capture the ordering of the incorporated attribute vector in the image grid. Due to zero-mean normalization in Equation 1 by subtraction the sample mean μ the descriptor is invariant to photometric and illumination changes.

Tuzel et al. [19] applied integral images to efficiently compute the mean and the covariance descriptor for rectangular regions. Using the law of total variation the intermediate computation of sums and squared sums within the specified regions results in the final covariance matrix representation. Since covariance matrices are symmetric, an overall number of $d + d(d+1)/2$ integral images enables the full construction. d integral images provide the information for the mean computation, while the remaining summed area tables include the tensors of each permuted pair of the input channels. Integral images offer to extract summed values in a given region in constant time, independently of the region size. The detailed construction and the implementation issues can be found in [19].

3.2 Approximated Representation and Distance Computation on Euclidean Space

Similarity measurement computations between second order moments in high dimensions such as the *Foerstner* distance suffer from expensive eigenvalue computations and manifold mappings. However, in real-time applications such as tracking and large scale computations speed poses an important issue. The *Foerstner* metric used in [6, 17] defines a symmetric distance approximation between raw covariance matrices through $\log$-manifold mappings and can be obtained by computing the sum of squared logarithms of the generalized eigenvalues λ_i:

$$d_f(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_i \ln \lambda_i(\mathbf{A}, \mathbf{B})}, \tag{3}$$

where $\mathbf{A} \in \mathbf{R}^{dxd}$ and $\mathbf{B} \in \mathbf{R}^{dxd}$ denote two given symmetric positive semi-definite matrices. The definition of this metric guarantees symmetry and positivity similar to the properties of the Euclidean distance. However, the complexity is increased due to the $\log$-mappings of matrices to Lie space and of matrix inverses computations.

In this work and contrary to the *Foerstner* metric, where the similiarity measurement is approximated, we aim to approximate the individual first and second order moments, describing then Euclidean vector space. The idea is based on choosing a representative set of samples of two given distributions and to compute Euclidean distance for similarity. Moreover, the approximation of single covariance matrices describing Euclidean space enables significant simplifications in further processing such as the particle filtering and the update strategy.

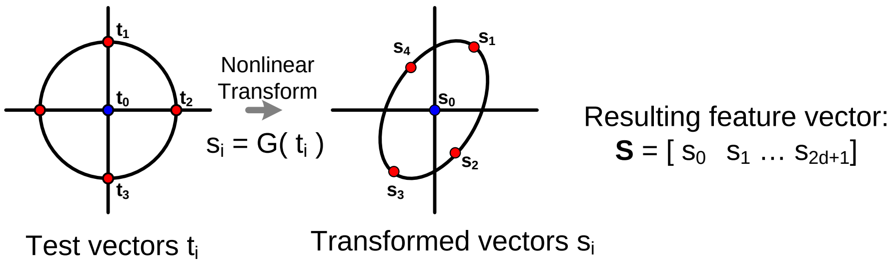


Figure 1: The non-linear transform of a given set of test vectors $\mathbf{t}_i$ to the second coordinate system, representing the properties of $G(\cdot)$. The resulting feature vector s captures both, mean and covariance information.

Julier et al. [9, 10] proposed the unscented transform (UT), which approximates a distribution by specified sampling instead of approximating an arbitrary non-linear function by mapping to manifolds. The UT provides an efficient estimator for probability distributions and was successfully applied to unscented *Kalman* filtering [12], where it overcomes the drawbacks of Taylor expansions truncated after the second order terms. In the d-dimensional case the UT relies on choosing a set of $2d + 1$ specific vectors $\mathbf{t}_i$. In unscented *Kalman* filtering [11] these d-dimensional test vectors $\mathbf{t}_i$ are propagated through the non-linear system and give an accurate estimation of the posterior mean and the covariance for any non-linearity. In contrast to *Monte Carlo* methods, where these test vectors are selected randomly, the points deterministically sample the intersection of the unit sphere and the d-dimensional Cartesian coordinate system together with the origin representing the mean. Given these test vectors $\mathbf{t}_i$, a non-linear transform $\mathbf{s}_i = G(\mathbf{t}_i)$ individually generates a new set of sample vectors $\mathbf{s}_i$ representing the properties of $G(\cdot)$ in a second coordinate system as shown in Figure 1. By computing the statistics of these points $\mathbf{s}$ the original mean and covariance information about $G(\cdot)$ up to second order [10] is captured. Figure 1 illustrates the specified sampling of the test vectors for a 2D case, the non-linear transform and depicts the construction of the Euclidean vector space representation.

According to the concept of [10] the non-linear transform $G = \sqrt{\alpha \mathbf{\Sigma}_r}$ generates the set of $2d + 1$ vectors $\mathbf{s}_i$ as follows:

$$s_0 = \mu \qquad s_i = \mu + (\sqrt{\alpha \mathbf{\Sigma}_r})_i \qquad s_{i+d} = \mu - (\sqrt{\alpha \mathbf{\Sigma}_r})_i, \tag{4}$$

where $i = 1 \dots d$ and $(\sqrt{\alpha \mathbf{\Sigma}_r})_i$ defines the i-th column of the square root matrix of $\mathbf{\Sigma}_r$. The scalar α denotes a constant weighting factor for the elements in the covariance matrix and is set to $\alpha = 2$ in case of Gaussian distribution [10]. Assuming a Gaussian distribution $N(\mu, \mathbf{\Sigma}_r)$ with mean μ and covariance matrix $\mathbf{\Sigma}_r$, we extract a specified set of vectors $\mathbf{s}_i$ providing the approximated sample covariance $\mathbf{\Sigma}'_r$ from the columns of matrix square root. Including μ in the generated set of vectors as a simple offset yields again the approximated original distribution $N(\mu', \mathbf{\Sigma}'_r)$ [10] according to

$$\mu \approx \mu' = \frac{1}{2d+1} \sum_{i=0}^{2d} \mathbf{s}_i, \tag{5}$$

$$\mathbf{\Sigma}_r \approx \mathbf{\Sigma}'_r = \frac{1}{2d} \sum_{i=0}^{2d} (\mathbf{s}_i - \mu') (\mathbf{s}_i - \mu')^T . \tag{6}$$

Due to symmetry and positive semi-definiteness of covariance matrices, the efficient *Cholesky* factorization can be applied to decompose $\mathbf{\Sigma}_r$ into LL^T, where L is lower triangular. In principle any method for matrix square root factorization can be used, however, the *Cholesky* decomposition provides a complexity of $O(n^3/3)$ with lowest number of operations for symmetric and semi-positive definite matrices.

It is obvious that each of these generated vectors $\mathbf{s}_i \in \mathbf{S}$ describes a d-dimensional Euclidean space, therefore, L^2 distance computations can be applied for similarity measurements. We concatenate the unweighted sample vectors to a resulting vector $\mathbf{S} \in \mathbf{R}^{d(2d+1)}$ capturing the original mean μ and the covariance matrix $\mathbf{\Sigma}_r$. Following the exact directive in approximating various distribution, the distance measurements between these vectors fulfill positiveness and symmetry. In the following section we show how this approximated representation can be applied to tracking.

4 Application to Tracking

The derived approximated covariance representation in Euclidean vector space enables a plausible and simple integration into the principle working step during tracking, like feature evaluation and online update strategies. In order to demonstrate the use of our approximated covariance representation, we briefly highlight these steps. First, we review the approach of particle filters for tracking, using several sub-parts to increase robustness of tracking. Second, we additionally evaluated the uncertainty of the particle set depending on the individual sub-parts, to dynamically weight the influence of each sub-part for the tracking result. Third, we describe the features used for tracking. Finally, the incremental PCA (iPCA) for efficient appearance representation is maintained for each sub-part. While the integration of the uncertainty values incorporates knowledge about the surrounding, the iPCA creates a optimal representation for each sub-part.

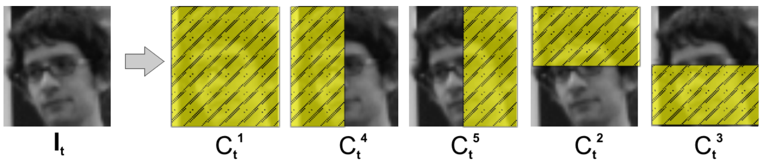


Figure 2: Subdivision of the tracking object. Individual representations for each sub-parts are maintained during the tracking process.

4.1 Particle Filter for Efficient State Estimation

Particle filtering for tracking [8] provides a probabilistic framework, that maintains multiple hypotheses of the current object state and has been proved to yield impressively robust tracking results. The probability distribution of the hidden target state $\mathbf{x}_t$ of the tracked object at time step t is estimated using a set of N_P weighted particles $\{x_t^i, w_t^i\}$ with $i = 0...N_P$ and associated measurements z_t^i. Each particle x_t^i simulates the real hidden state of the the object. Using the dynamic model $p(x_t^i|x_{t-1}^i)$ and the observation likelihood $p(z_t^i|x_t^i)$, the posterior distribution $p(\mathbf{x}_t|\mathbf{z}_t)$ is approximated by the finite set of particles:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{i=1}^{N_p} w_t^i x_t^i \quad where \quad w_t^i \propto w_{t-1}^i \frac{p(z_t^i \mid x_t^i) p(x_t^i, \mid x_{t-1}^i)}{q(x_t^i \mid x_{t-1}^i, z_t^i)} \tag{7}$$

where $\sum_{i=1}^{N_p} w_t^i = 1$ is fulfilled and $q(x_t^i \mid x_{t-1}^i, z_t^i)$ is the proposal distribution from which the particles are drawn. Using the state transition model $p(x_t^i|x_{t-1}^i)$ as proposal distribution leads to the bootstrap filter, where the weights are directly proportional to the observation model $p(z_t^i|x_t^i)$. Finally, the posterior density $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ is approximated by the weighted mean over the particle distribution, as given in Equation 7. To avoid the degeneracy of the particle set, the resampling of the weights is performed after each frame. (See [1] for more details.) To increase the robustness of tracking and to handle the tracking procedure during occlusions, we divide the object representation into five sub-parts. The partition into the five sub-parts is shown in Figure 2. Each sub-part is represented by a reference feature vector $\mathbf{S}_{j,ref}, j \in \{1..5\}$, and updated independently during tracking and the importance of each part is adapted individually.

4.2 Uncertainty of Particle Sets

In the case of background clutter, occlusion, or ambiguities the distribution of the particles can become unsubstantial, which leads to drifting, inaccurate tracking results or lost objects. From this it follows that an important task is to measure the quality of a given weighted particle set $\{x_t^i, w_t^i\}$, to recognize if the tracking of objects fail.

One method for measuring this uncertainty has been proposed by Maggio et al. [15], based on the analysis of the covariance matrix Σ_w of the weighted particles $\{x_t^i, w_t^i\}$. The determinant of Σ_w describes the volume of the hyper-ellipse in the state space, given by the product of the eigenvalues of Σ_w, defining the uncertainty $U_C = det(\Sigma_w) = \prod_{k=1}^{d} \lambda_k$, where d is the dimensionality of the state vector x^i. Maggio et al. [15] calculated several uncertainties for particle weights from different feature cues and used their proportions to dynamically adjust their influence onto the tracking result. Recently, Badrinarayanan et al. [3] improved this measurement by using the ratio between the determinants of Σ_w and Σ_s, where Σ_s depicts the covariance of the particles without weighting by the likelihood $\left\{x_t^i, \frac{1}{N_p}\right\}$. They pointed out that this increases the stability of the uncertainty measurement regarding the fluctuations in the particle spread over time. They finally defined the uncertainty of a particle set as $U_c = \min\left\{1.0, \frac{\Sigma_w}{\Sigma_s}\right\}$, which is near 0 for a peaked particle distribution and going to 1 for an uncertain result. We indent to use this uncertainty measurement as an additional weighting term for the influence of each part in the final state. The weighting for part j in frame t is defined as

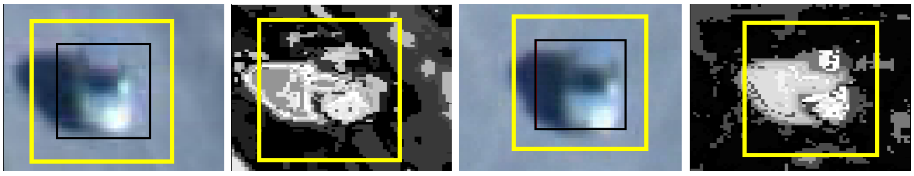
$$w_u^j = (1 - \min\left\{1.0, \frac{\Sigma_{j,w}}{\Sigma_{j,s}}\right\}), \quad where \quad \sum_{j=1}^{5} w_u^j = 1. \tag{8}$$

Finally, the observation likelihood for particle i can be computed by

$$p(z_t^i \mid x_t^i) \propto exp(-\frac{err_i}{\sigma}) \quad with \quad err_i = \sum_{j=1}^{5} w_u^j \, \|\mathbf{S}_j - \mathbf{S}_{j,ref}\|^2. \tag{9}$$

4.3 Features for Tracking

Due to efficient computation of the covariance feature representation, we include several features to capture spatial *(x,y)*, color *(R,G,B)* and texture information *(*$|I_x(x,y)|$, $|I_y(x,y)|$, $|I_{xx}(x,y)|$, $|I_{yy}(x,y)|$*)*



a) b)

Figure 3: a) Foreground probability after initialization. The dark rectangle (black) represents the foreground region, while the bright (yellow) rectangle also includes information about the background. b) Resulting foreground probability map after several update steps during tracking.

similar to [17]. Additionally, the feature vector includes the foreground probability $p_F((x,y) \in \mathrm{O}|I(x,y) \equiv b)$ for each pixel. The rectangular selection of the object-of-interest includes background pixels, which have to be removed in a first step by classifying every pixel into foreground or background.

The ratio between the non-normalized histograms of the object patch $\mathbf{H}_o(b)$ and its surrounding $\mathbf{H}_s(b)$, including the object patch as well, $\mathbf{H}_o(b) \subset \mathbf{H}_s(b)$, defines the probability of a pixel at (x,y) to be part of the object O given by $p_F((x,y) \in \mathrm{O}|I(x,y) \equiv b) = \frac{\mathbf{H}_o(b)+1}{\mathbf{H}_s(b)+2}$, where $I(x,y)$ is the color of the pixel at position (x,y) and b is the assigned histogram bin.

The foreground/background histograms are iteratively updated during the tracking process. In the case of static cameras foreground/background information can also be used to favor moving objects for tracking. In Figure 3 the initialization process and resulting probabilities after several updates are demonstrated. The dark rectangle (black) depicts the foreground region, while the bright (red) rectangle also includes information about the background.

The resulting ten-dimensional feature vector f is defined by

$$f = \left[\begin{array}{cccccccccc} x & y & R & G & B & |I_x(x,y)| & |I_y(x,y)| & |I_{xx}(x,y)| & |I_{yy}(x,y)| & p_F(x,y) \end{array} \right] \quad (10)$$

4.4 Incremental Subspace Learning for Part-based Representation

The on-line adaption of the object representation is of vital interest during visual tracking. Although covariance based features are in general robust against intensity changes, an on-line learning of a subspace representation is applied in this work to further handle variations. We use the incremental PCA method proposed by [18]. To obtain an initial model, a batch-based PCA is applied to a small set of k training vectors $\mathbf{S} \in \mathbf{R}^n$ sampled around the initialization position of an object, where n denotes the vector dimension of the approximated feature representation according to $n = d(2d+1)$. Furthermore, this results in the initial eigenspace $\mathbf{U}_0 \in \mathbf{R}^{nxm}$, where m depicts the number of eigenvectors, coefficient matrix $A_0 \in \mathbf{R}^{mxk}$ and mean vector $\mu_0 \in \mathbf{R}^n$. With a new feature vector arriving in the next frame the eigenspace representation gets updated, and the dimension of the eigenspace is increased by one. To preserve the dimension of the subspace the least significant eigenvector is discarded. The advantage of the method proposed by [18] is that previous samples need not to be stored and the only required information are the coefficients and eigenvalues. For tracking, the incremental subspace is updated independently for each of the five object parts. The dimensionality of the eigenspace is determined during the initialization of the tracker, depending on the eigenvalues returned from the subspace. In addition to the sensitivity to the background, achieved by incorporating the uncertainty measurement, the eigenspaces create an optimal representation of the object in each sub-part. The likelihood computation (Equation 9) changes to the reconstruction error of the feature vector $\mathbf{s}$ given by

$$p(z_t^i \mid x_t^i) \propto \exp(-\frac{rep_i}{\sigma}) \quad with \quad rep_i = \sum_{j=1}^{5} w_u^j \left\|(\mathbf{S}_j - \mu_j) - \mathbf{U}_j \mathbf{U}_j^T (\mathbf{S}_j - \mu_j)\right\|^2. \tag{11}$$

5 Experimental Evaluation

In order to demonstrate the tracking capability, our approaches is evaluated on several gray scale and colored video sequences, mainly publicly available such as PETS01, Vivid PETS05 and Caviar. Different characteristics like challenging variations in lighting and camera positions, occlusions, similar objects, scale, and moving cameras are included. To show the performance of our general approach the configuration for all video scenarios is kept fixed. For all experiments we apply a particle filter with 500 particles, representing a four-dimensional state space $\mathbf{x}_t = [x, y, f_w, f_h]$ for scaling with parameter f by keeping the aspect ratio of the object constant, and a random walk model with no transition prediction. The feature vectors $\mathbf{s}$ used for the observation likelihood are given in Equation 10 and are reduced to a dimension of 8 for the experiments with gray scale sequences. Foreground/background estimation is based on 10 bins per color channel.

In our first experiment we compare the part based tracking using the Euclidean vector space approximation of the covariance matrices (Section 3.2) to the method proposed by Porikli et al. [17]. Following [17], we apply an exhaustive search to extract the covariance representation on every pixel location, followed by a subsequent mean-shift procedure to obtain the final tracking result. The update strategy is performed on a Riemannian manifold, incorporating the last 20 frames, as proposed in [17]. Figure 4 shows the obtained results of our proposed part-based tracker using Euclidean approximation of the covariance matrices (dashed rectangles) compared to the method in [17] (solid rectangles). Obviously, the global object description suggested in [17] is more sensitive to distractions from the background, especially homogeneous regions, or similar objects. One can see that the approximation does not deteriorate the tracking performance. Rather the sub-part representation, together with the dynamically adjustment of the rejection probability for each sub-part over time, allows our tracker to handle critical situations during mutual occlusions and overlaps.

As a second experiment, we present the improved results obtained by using the eigenspace representation (Section 4.4) for each sub-part. Figure 5 depicts several examples from publicly available scenes. The integration of the incremental PCA in the object representation results in more stable results, concerning scale changes and drifting over time. Additionally, the results show the capability of our approximated covariance representation for standard state-of-the-art machine learning techniques.

6 Conclusions

In this paper we have demonstrated a powerful method for approximation of first and second order moments such as mean vectors and covariance matrices for the application of tracking. Based on efficient integral structures for the covariance matrix computation, we applied the *Cholesky* factorization to obtain an approximated feature representation with integrated mean information. Due to a Euclidean vector space representation of the approximation, we have demonstrated that costly similarity

Figure 4: Visual tracking results of the covariance descriptor proposed in [17] (solid) compared to our part based tracker using Euclidean approximation of covariance matrices (dashed).

measurements on manifolds can be replaced by simple distance computations in higher dimensional Euclidean space. Furthermore, the covariance approximation enables processing steps such as incremental sub-space model update strategies directly on Euclidean space. In the experimental section we compared our approach to common covariance based tracking methods and demonstrated robustness by incorporating incremental PCA in the object representation. Future work will include an improved on-line learning and update strategy and an integrated feature selection procedure using binary masking of the approximated vector representation. A full C/C++ implementation will further improve the real-time performance of currently 6-8 fps in Matlab, using mex-files for feature computation and representation.

Acknowledgment

This work was supported by the Austrian Science Fund Projects P18600 and W1209 under the doctoral program Confluence of Vision and Graphics and the APAFA Project No. 813397, financed by the Austrian Research Promotion Agency (www.ffg.at). We want to thank Dr. Peter M. Roth for many discussions, his advices, and support in the field of the subspace methods.

References

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

[2] Shai Avidan. Ensemble tracking. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

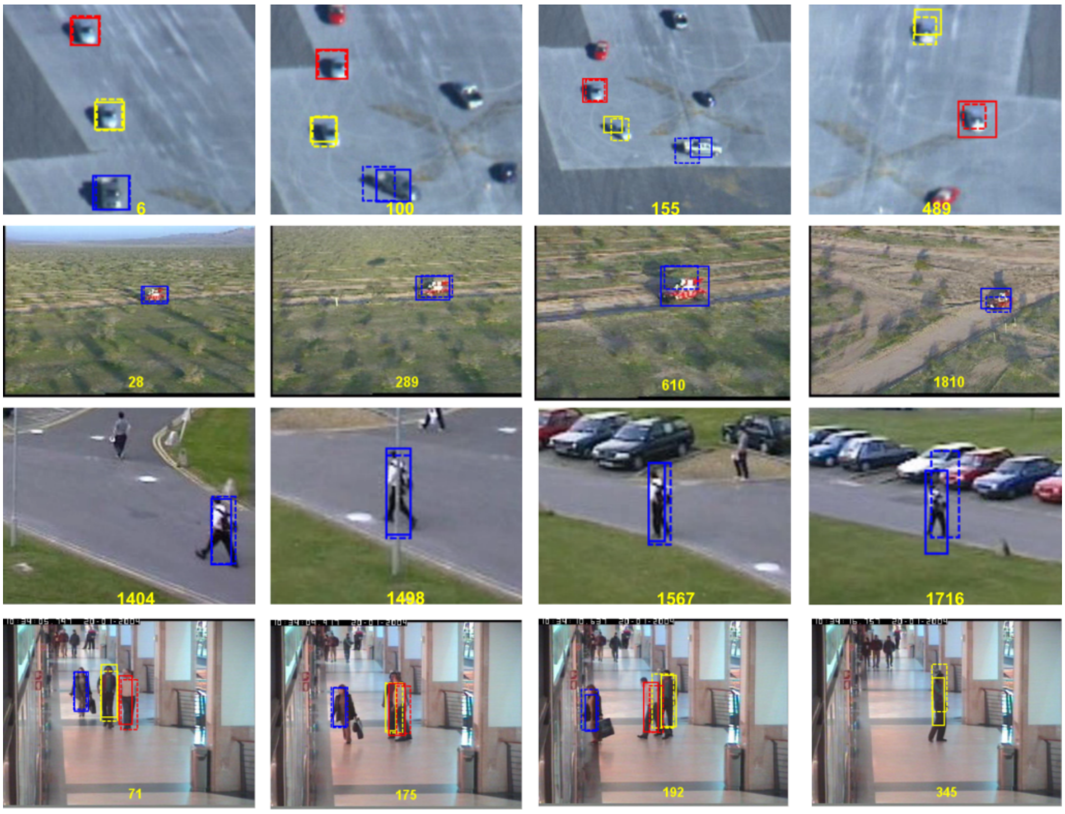


Figure 5: Improvements for the proposed sub-parts tracker (dashed) using the incremental eigenspace representations for each sub-part (solid).

[3] V. Badrinarayanan, P. Pérez, F. Le Clerc, and L. Oisel. Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues. In *Proceedings International Conference on Computer Vision*, 2007.

[4] Stan Birchfield and Sriram Rangarajan. Spatiograms versus histograms for region-based tracking. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[5] D. Comaniciu, Ramesh V., and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[6] W. Foerstner and B. Moonen. A metric for covariance matrices. Technical report, Dept. of Geodesy and Geoinformatics, Stuttgart University, 1999.

[7] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via on-line boosting. In *Proceedings British Machine Vision Conference*, 2006.

[8] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proceedings European Conference on Computer Vision*, 1996.

[9] Simon Julier. The scaled unscented transformation. In *Proceedings of the American Control Conference*, pages 4555–4559, 2002.

[10] Simon Julier and Jeffrey K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. Technical report, 1996.

[11] Simon Julier and Jeffrey K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls, Orlando, FL*, 1997.

[12] Rudolf E. Kálmán. A new approach to linear filtering and prediction problems. *Trans. of the ASME Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[13] Xi Li, Weiming Hu, Zhongfei Zhang, Xiaoqin Zhang, Mingliang Zhu, and Jian Cheng. Visual tracking via incremental log-euclidean riemannian subspace learning. In *Proceedings IEEE Conference Computer Vision and Pattern Recognition*, 2008.

[14] Jongwoo Lim, David Ross, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for visual tracking. In *Neural Information Processing Systems*, pages 793–800, 2004.

[15] E. Maggio, F. Smerladi, and A. Cavallaro. Combining colour and orientation for adaptive particle filter-based tracking. In *Proceedings British Machine Vision Conference*, 2005.

[16] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *Proceedings European Conference on Computer Vision*, 2002.

[17] F. Porikli, O. Tuzel, and P. Meer. Covariance tracking using model update based on lie algebra. In *Proceedings IEEE Conference Computer Vision and Pattern Recognition*, 2006.

[18] Danijel Skočaj and Aleš Leonardis. Weighted and robust incremental method for subspace learning. In *Proceedings International Conference on Computer Vision*, 2003.

[19] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. In *Proceedings European Conference on Computer Vision*, 2006.

[20] O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifolds. In *Proceedings IEEE Conference Computer Vision and Pattern Recognition*, 2007.

[21] O. Tuzel, F. Porikli, and P. Meer. Learning on lie groups for invariant detection and tracking. In *Proceedings IEEE Conference Computer Vision and Pattern Recognition*, 2008.

[22] Ambrish Tyagi and James W. Davis. A recursive filter for linear systems on riemannian manifolds. In *Proceedings IEEE Conference Computer Vision and Pattern Recognition*, 2008.

[23] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2002.

[24] Hanzi Wang, David Suter, and Konrad Schindler. Effective appearance model and similarity measure for particle filtering and visual tracking. In *Proceedings European Conference on Computer Vision*, 2006.

[25] Jian Yao and Jean-Marc Odobez. Fast human detection from videos using covariance features. In *Proceedings European Conference on Computer Vision*, 2008.