Fast and Robust Appearance-based Tracking

Stephan Liwicki, Stefanos Zafeiriou, Georgios Tzimiropoulos and Maja Pantic

Abstract— **We introduce a fast and robust subspace-based approach to appearance-based object tracking. The core of our approach is based on Fast Robust Correlation (FRC), a recently proposed technique for the robust estimation of large translational displacements. We show how the basic principles of FRC can be naturally extended to formulate a robust version of Principal Component Analysis (PCA) which can be efficiently implemented incrementally and therefore is particularly suitable for robust real-time appearance-based object tracking. Our experimental results demonstrate that the proposed approach outperforms other state-of-the-art holistic appearance-based trackers on several popular video sequences.**

I. Introduction

Visual tracking in unconstrained environments is an unsolved problem. For example, in real-world face analysis applications, tracking algorithms have to deal with significant appearance changes induced by sudden head motions, non-rigid facial deformations as well as illumination changes, cast shadows and occlusions. Such phenomena typically make most existing tracking algorithms fail.

The appearance-based approach to tracking has been one of the *de facto* choices for tracking faces in image sequences. Prominent examples of such an approach include subspace-based techniques [1], mixture models [2], [3], discriminative models for regression/classification [4], gradient descent [5] and very often combinations of the above [1], [6]–[10]. In this paper, we propose a subspace-based tracking algorithm which, to some extend, is able to provide a remedy to typical problems encountered in face analysis applications by featuring many favorable properties. Our algorithm is closely related to the incremental visual tracker (IVT) of Ross *et al.* [9] and its incremental kernel PCA (IKPCA) extension proposed by Chin and Suter [10], and as such can deal with drastic appearance changes, does not require offline training, continually updates a compact object representation and uses the Condensation algorithm [11] to robustly estimate the object's location.

Similarly to IVT and IKPCA, our method is essentially an eigentracker [1] where the eigenspace is adaptively learned and updated online. The key element which makes our approach equally fast but significantly more robust, is how the eigenspace is generated. Ross *et al.* use standard ℓ_2 norm PCA. Unfortunately, the ℓ_2 norm enjoys optimality properties only when image noise is independent and identically distributed (i.i.d.) Gaussian; however, for data corrupted by outliers, the estimated subspace can be arbitrarily skewed [12]. A somewhat more robust approach is the method proposed by Chin and Suter which incrementally learns a non-linear subspace via KPCA [10]. The tracking process requires the computation of the pre-images which imposes a trade-off between efficiency and robustness while experimental results show that the gain in robustness appears to be not very significant.

On the contrary, the proposed tracker is based on a robust reformulation of PCA which requires straightforward optimizations and is as computationally efficient as ℓ_2 norm PCA. More specifically, our approach is based on a dissimilarity measure originally introduced by Fitch *et al.* in the context of robust correlation-based estimation of large translational displacements [13]. The basic idea is to suppress gross errors by encoding pixel intensities as angles and measure dissimilarity using the cosine of angle differences. We show how the framework for robust correlation can be naturally extended to form a robust version of PCA which replaces the ℓ_2 norm with the dissimilarity measure of Fitch *et al.*. Finally, we use our direct robust PCA within the framework of IVT for efficient and robust appearance-based tracking.

II. Fast, Direct and Robust PCA

A. Principal Component Analysis with ℓ_2 Norm

Let $\mathbf{x}_i$ be the d-dimensional vector obtained by writing image I_i in lexicographic ordering. We assume that we are given a population of n samples $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_n] \in \mathbb{R}^{d \times n}$. Let us also denote by $\overline{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$ and $\overline{\mathbf{X}}$ the sample mean and the centralized sample matrix of $\mathbf{X}$. ℓ_2 norm PCA finds a set of $p < d$ (usually, $p \ll d$) orthonormal basis functions $\mathbf{B} = [\mathbf{b}_1 \cdots \mathbf{b}_p] \in \mathbb{R}^{d \times p}$ by minimizing the error function

$$e(\mathbf{B}) = ||\mathbf{X} - \mathbf{B}\mathbf{B}^T\mathbf{X}||_F^2 \tag{1}$$

where $||.||_F$ denotes the Frobenius norm. The above optimization problem is equivalent to:

$$\begin{aligned} f(\mathbf{B}) &= \text{tr}\left[\mathbf{B}^T\overline{\mathbf{X}}\,\overline{\mathbf{X}}^T\mathbf{B}\right] \\ &\text{subject to } \mathbf{B}^T\mathbf{B} = \mathbf{I} \end{aligned} \tag{2}$$

where tr[.] is the trace of a matrix. The solution is given by the eigenvectors corresponding to the p largest eigenvalues obtained from the eigendecomposition of the covariance matrix $\mathbf{S} = \overline{\mathbf{X}}\,\overline{\mathbf{X}}^T$ (or the Singular Value Decomposition (SVD) of $\overline{\mathbf{X}}$). Finally, the reconstruction of $\mathbf{X}$ from the subspace spanned by the columns of $\mathbf{B}$ is given by $\tilde{\mathbf{X}} = \mathbf{B}\mathbf{C} + \mathbf{M}$, where $\mathbf{C} = \mathbf{B}^T\overline{\mathbf{X}}$ is the matrix with the set of

S. Liwicki, S. Zafeiriou, G. Tzimiropoulos and M. Pantic are with the Department of Computing, Imperial College London, United Kingdom. Contact: sl609@imperial.ac.uk

M. Pantic is also with the Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, The Netherlands.

projection coefficients and $\mathbf{M}$ is a matrix with n columns each of which is the mean vector $\overline{\mathbf{x}}$.

B. Cosine-based Error Function

The error function in (1) is based on the ℓ_2 norm and therefore is extremely sensitive to gross errors caused by outliers [12]. Motivated by the recent work of Fitch *et al.* on robust correlation-based translation estimation [13], we replace the ℓ_2 norm with the following dissimilarity measure

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{d} \{1 - \cos(\alpha\pi[\mathbf{x}_i(k) - \mathbf{x}_j(k)])\} \qquad (3)$$

where the pixel values of the corresponding images I_i, I_j are represented in the range $[0, 1]$ and $\alpha \in \mathbb{R}^+$.

As noted by Fitch *et al.*, for pixel intensities in the range $[0, 1]$, (3) is equivalent to Andrews' M-Estimate [13]. In particular, Andrews' influence function, *i.e.* the derivative of a kernel, is given by

$$\psi(r) = \begin{cases} \sin(\pi r) & \text{if } \; -1 \leq r \leq 1 \\ 0 & \text{otherwise.} \end{cases} \qquad (4)$$

The Fast Robust Correlation (FRC) scheme proposed by Fitch *et al.* [13] utilizes (3) and, unlike ℓ_2-based correlation, is able to estimate large translational displacements in real images while achieving the same computational complexity. In the following, we show how to exploit the cosine kernel to formulate a direct robust version of PCA.

C. Fast, Direct and Robust PCA

To show how (3) can be used as a basis for direct and robust PCA, for notational convenience, let us first define $\boldsymbol{\theta}_i \triangleq \alpha\pi\mathbf{x}_i$, $\cos\boldsymbol{\theta}_i \triangleq [\cos\boldsymbol{\theta}_i(1) \cdots \cos\boldsymbol{\theta}_i(d)]^T$ and $\sin\boldsymbol{\theta}_i \triangleq [\sin\boldsymbol{\theta}_i(1) \cdots \sin\boldsymbol{\theta}_i(d)]^T$. We also assume that $\mathbf{x}_i(k)$ is in the range $[0, 1]$. We have

$$\begin{aligned} d(\mathbf{x}_i, \mathbf{x}_j) &= \sum_{k=1}^{d} \{1 - \cos(\boldsymbol{\theta}_i(k) - \boldsymbol{\theta}_j(k))\} \\ &= d - \sum_{k=1}^{d} \{\cos\boldsymbol{\theta}_i(k)\cos\boldsymbol{\theta}_j(k) \\ &+ \sin\boldsymbol{\theta}_i(k)\sin\boldsymbol{\theta}_j(k)\} \\ &= d - \begin{bmatrix} \cos\boldsymbol{\theta}_i \\ \sin\boldsymbol{\theta}_i \end{bmatrix}^T \begin{bmatrix} \cos\boldsymbol{\theta}_j \\ \sin\boldsymbol{\theta}_j \end{bmatrix} \\ &= d - \begin{bmatrix} \cos(\alpha\pi\mathbf{x}_i) \\ \sin(\alpha\pi\mathbf{x}_i) \end{bmatrix}^T \begin{bmatrix} \cos(\alpha\pi\mathbf{x}_j) \\ \sin(\alpha\pi\mathbf{x}_j) \end{bmatrix} \\ &= ||\mathbf{z}_i - \mathbf{z}_j||^2. \end{aligned} \qquad (5)$$

The last equality makes the basic computational module of the proposed scheme apparent. That is, we define the mapping from $[0, 1]$ to the $(2d)$-dimensional sphere with radius $\sqrt{d}$

$$\mathbf{z}_i = \frac{1}{\sqrt{2}} \begin{bmatrix} \cos(\alpha\pi\mathbf{x}_i) \\ \sin(\alpha\pi\mathbf{x}_i) \end{bmatrix} \qquad (6)$$

and apply linear PCA to the transformed data. Notice that when $\alpha < 2$, this mapping is one-to-one and, therefore, reconstruction of the original input space is feasible by applying simple trigonometry.

```
Algorithm 1 ESTIMATING THE PRINCIPAL SUBSPACE
Input: A set of $n$ images $I_i$, $i = 1, \ldots, n$, of $d$ pixels, the number $p$ of principal components and parameter $\alpha$.
Output: The principal subspace $\mathbf{B}$, eigenvalues $\boldsymbol{\Sigma}$ and mean vector $\overline{\mathbf{z}}$ of the transformed data.
Step 1. Represent $I_i$ in $[0, 1]$ and obtain $\mathbf{x}_i$ by writing $I_i$ in lexicographic ordering.
Step 2. Compute $\mathbf{z}_i$ using (6), form the matrix of the transformed data $\mathbf{Z} = [\mathbf{z}_1 \cdots \mathbf{z}_n] \in \mathbb{R}^{2d \times n}$ and compute $\overline{\mathbf{z}}$ and the centralized sample matrix $\overline{\mathbf{Z}}$.
Step 3. Compute the matrix $\mathbf{W} = \overline{\mathbf{Z}}^T\overline{\mathbf{Z}} \in \mathbb{R}^{n \times n}$ and find the eigendecomposition of $\mathbf{W} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$.
Step 4. Find the $p$-reduced set, $\mathbf{U}_p \in \mathbb{R}^{2d \times p}$ and $\boldsymbol{\Lambda}_p \in \mathbb{R}^{p \times p}$.
Step 5. Compute $\mathbf{B} = \mathbf{Z}\mathbf{U}_p\boldsymbol{\Lambda}_p^{-\frac{1}{2}} \in \mathbb{R}^{2d \times p}$ and $\boldsymbol{\Sigma} = \boldsymbol{\Lambda}^{\frac{1}{2}}$.
Step 6. Reconstruct using $\tilde{\mathbf{Z}} = \mathbf{B}\mathbf{B}^T\overline{\mathbf{Z}} + \mathbf{M}$, where $\mathbf{M}$ contains the mean vector $\overline{\mathbf{z}}$ as columns.
Step 7. Go back to the pixel domain using trigonometry.
```

```
Algorithm 2 EMBEDDING OF NEW SAMPLES
Input: An image $J$ of $d$ pixels and the principal subspace B of Algorithm 1.
Step 1. Represent $J$ in $[0, 1]$ and obtain $\mathbf{y}$ by writing $J$ in lexicographic ordering.
Step 2. Find $\mathbf{z}$ using (6) and obtain embedding as $\mathbf{B}^T\mathbf{z}$.
```

For high-dimensional data such as images, the proposed framework enables a fast implementation by making use of the following theorem [14].

Theorem I: Define matrices $\mathbf{A}$ and $\mathbf{B}$ such that $\mathbf{A} = \boldsymbol{\Phi}\boldsymbol{\Phi}^T$ and $\mathbf{B} = \boldsymbol{\Phi}^T\boldsymbol{\Phi}$. Let $\mathbf{U}_A$ and $\mathbf{U}_B$ be the eigenvectors corresponding to the non-zero eigenvalues $\boldsymbol{\Lambda}_A$ and $\boldsymbol{\Lambda}_B$ of $\mathbf{A}$ and $\mathbf{B}$, respectively. Then, $\boldsymbol{\Lambda}_A = \boldsymbol{\Lambda}_B$ and $\mathbf{U}_A = \boldsymbol{\Phi}\mathbf{U}_B\boldsymbol{\Lambda}_A^{-\frac{1}{2}}$.

Algorithm 1 summarizes the steps of our direct robust PCA. Our framework also enables the direct embedding of new samples. **Algorithm 2** summarizes this procedure.

D. A Kernel PCA Perspective

The proposed PCA with the cosine-based dissimilarity measure can be interpreted as a kernel PCA (KPCA). Let $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a positive definite function that satisfies the Mercer's conditions. Then, k defines an arbitrary dimensional Hilbert space $\mathcal{H}$ (the so-called feature space in the rest of the paper) through an implicit mapping $\phi : \mathbb{R}^d \to \mathcal{H}$ such that $k(\mathbf{x}_i, \mathbf{x}_j) = \langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle$. KPCA [15] is defined exactly as PCA in feature space and aims at finding a set of projection bases by minimizing the least-squares reconstruction error in the feature space.

Let us define the kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2} \sum_{k=1}^{d} \cos(\alpha\pi[\mathbf{x}_i(k) - \mathbf{x}_j(k)]) \qquad (7)$$

Theorem II: The kernel defined in (7) is positive semi-definite.

```
Algorithm 3 INCREMENTAL PRINCIPAL SUBSPACE ESTIMATION
Input: A mean vector $\overline{\mathbf{z}}_n$, the principal subspace $\mathbf{B}_n \in \mathbb{R}^{2d\times p}$, the root of the corresponding eigenvalues $\mathbf{\Sigma}_n \in \mathbb{R}^{p\times p}$, a set of new images $\{I_{n+1},\dots,I_{n+m}\}$, the number $p$ of principal components and parameter $\alpha$.
Output: The new subspace $\mathbf{B}_{n+m}$, eigenvalues $\mathbf{\Sigma}_{m+n}$ and new mean $\overline{\mathbf{z}}_{n+m}$.
Step 1. From set $\{I_{n+1},\dots,I_{n+m}\}$ compute the matrix of the transformed data $\mathbf{Z}_m = [\mathbf{z}_{n+1} \cdots \mathbf{z}_{n+m}]$ and the mean vector $\overline{\mathbf{z}}_m$.
Step 2. Compute the new mean vector $\overline{\mathbf{z}}_{n+m} = \frac{n}{n+m}\overline{\mathbf{z}}_n + \frac{m}{n+m}\overline{\mathbf{z}}_m$ and form matrix
    $\mathbf{F} = \left[(\mathbf{z}_{n+1} - \overline{\mathbf{z}}_m) \cdots (\mathbf{z}_{m+n} - \overline{\mathbf{z}}_m)\ \sqrt{\frac{nm}{n+m}}(\overline{\mathbf{z}}_m - \overline{\mathbf{z}}_n)\right]$.
Step 3. Compute $\tilde{\mathbf{F}} = \text{orth}(\mathbf{F} - \mathbf{B}_n{\mathbf{B}_n}^T\mathbf{F})$ and
    $\mathbf{R} = \begin{bmatrix} \mathbf{\Sigma}_n & {\mathbf{B}_n}^T\mathbf{F} \\ \mathbf{0} & \tilde{\mathbf{F}}(\mathbf{F} - \mathbf{B}_n{\mathbf{B}_n}^T\mathbf{F}) \end{bmatrix}$ (where orth(.) performs orthogonalization).
Step 4. Compute $\mathbf{R} \overset{svd}{=} \tilde{\mathbf{B}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^T$ and obtain the $p$-reduced set $\tilde{\mathbf{B}}_p$ and $\tilde{\mathbf{\Sigma}}_p$.
Step 5. Compute $\mathbf{B}_{n+m} = [\mathbf{B}_n\ \tilde{\mathbf{F}}]\tilde{\mathbf{B}}_p$ and set $\mathbf{\Sigma}_{n+m} = \tilde{\mathbf{\Sigma}}_p$.
```

Proof: Using the analysis in (5), we can write the kernel $k(\mathbf{x}_i, \mathbf{x}_j)$ as a dot product:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{\sqrt{2}}\begin{bmatrix} \cos(\alpha\pi\mathbf{x}_i) \\ \sin(\alpha\pi\mathbf{x}_i) \end{bmatrix}^T \frac{1}{\sqrt{2}}\begin{bmatrix} \cos(\alpha\pi\mathbf{x}_i) \\ \sin(\alpha\pi\mathbf{x}_i) \end{bmatrix} \quad (8)$$

which proves **Theorem II**.

Using (7), we can write the proposed dissimilarity measure (3) as

$$\begin{aligned} d(\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)) &= ||\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)||^2 \\ &= k(\mathbf{x}_i, \mathbf{x}_i) - 2k(\mathbf{x}_i, \mathbf{x}_j) + k(\mathbf{x}_j, \mathbf{x}_j) \\ &= d - \textstyle\sum_{k=1}^{d} \cos(\alpha\pi[\mathbf{x}_i(k) - \mathbf{x}_j(k)]) \end{aligned} \quad (9)$$

Moreover, from (8) we can easily verify that $\phi(\mathbf{x}_i)$ has a closed form, *i.e.* $\phi(\mathbf{x}_i) = \mathbf{z}_i = \frac{1}{\sqrt{2}}\begin{bmatrix} \cos(\alpha\pi\mathbf{x}_i) \\ \sin(\alpha\pi\mathbf{x}_i) \end{bmatrix}$. This is in contrast to other popular kernels in machine learning, such as Gaussian RBFs [10], [15], for which ϕ is defined only implicitly. Such kernels allow only for inexact fast incremental versions of KPCA [10]. On the other hand, since in our case, the mapping is explicit, our incremental robust PCA is both fast and exact. **Algorithm 3** summarizes the main steps.

III. FAST AND ROBUST TRACKING

Similarly to Ross *et al.* [9], we model the tracking process using a Markov model with hidden states as affine transform $\mathbf{A}_t$. That is, the location of the object at time t is defined by the affine transform parameters $\mathbf{A}_t$. Given a set of observations $\mathbf{Z}_t = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$, $\mathbf{A}_t$ can be computed by maximizing $p(\mathbf{A}_t|\mathbf{Z}_t)$

$$p(\mathbf{A}_t|\mathbf{Z}_t) \propto p(\mathbf{z}_t|\mathbf{A}_t)\int p(\mathbf{A}_t|\mathbf{A}_{t-1})p(\mathbf{A}_{t-1}|\mathbf{Z}_{t-1})d\mathbf{A}_{t-1} \quad (10)$$

```
Algorithm 4 TRACKING ALGORITHM FOR TIME $t$
Input: Mean vector $\overline{\mathbf{z}}_{t-1}$, subspace $\mathbf{B}_{t-1}$, location $\mathbf{A}_{t-1}$ of time $t-1$ and current image frame $I_t$.
Step 1. Draw a number of particles $\mathbf{A}^p$ (in our case 600) from $p(\mathbf{A}_t|\mathbf{A}_{t-1})$.
Step 2. Take all image patches from $I_t$ which corresponds to particles $\mathbf{A}^p$ and order them lexicographically to form vectors $\mathbf{y}^p$ and compute $\mathbf{z}^p$ using (6).
Step 3. Choose $\{\mathbf{A}_t, \mathbf{z}_t\} = \arg\max_{\mathbf{A}^p, \mathbf{z}^p} p(\mathbf{z}^p|\mathbf{A}^p)$.
Step 4. Using $\mathbf{z}_t$ update mean and subspace by applying Algorithm 3.
```

To obtain an approximation for the above, we used a variant of the well-known Condensation algorithm [9], [11] using

- A dynamical model between states $p(\mathbf{A}_t|\mathbf{A}_{t-1})$
- A observation model $p(\mathbf{z}_t|\mathbf{A}_t)$

A. Modeling $p(\mathbf{A}_t|\mathbf{A}_{t-1})$

We used a typical Brownian motion model for modeling the dynamics between $\mathbf{A}_t$ and $\mathbf{A}_{t-1}$. That is, the elements of $\mathbf{A}_t$ are modeled independently by a Gaussian distribution around the previous state $\mathbf{A}_{t-1}$:

$$p(\mathbf{A}_t|\mathbf{A}_{t-1}) = \mathcal{N}(\mathbf{A}_t; \mathbf{A}_{t-1}, \mathbf{\Xi}) \quad (11)$$

where $\mathbf{\Xi}$ is a diagonal covariance matrix whose elements are the corresponding variances of the affine parameters. In a particle filtering fashion, we sample $p(\mathbf{A}_t|\mathbf{A}_{t-1})$ by drawing a number of particles from (11). It is well-known that there is a tradeoff between the number of particles, and how well the sampling approximates the distribution (11). In our experiments, we used 600 particles as in Ross *et al.* [9].

B. Modeling $p(\mathbf{z}_t|\mathbf{A}_t)$

Similarly to probabilistic PCA [16], we model the probability $p(\mathbf{z}_t|\mathbf{A}_t)$ as

$$p(\mathbf{z}_t|\mathbf{A}_t) = p_w(\mathbf{z}_t|\mathbf{A}_t)p_d(\mathbf{z}_t|\mathbf{A}_t) \quad (12)$$

where:

- $p_w(\mathbf{z}_t|\mathbf{A}_t)$ is the likelihood of the projected sample onto the principal subspace spanned by the columns of $\mathbf{B}$, modelled by the exponential of the Mahalanobis distance from the mean

$$p_w(\mathbf{z}_t|\mathbf{A}_t) = \mathcal{N}(\mathbf{z}_t; \overline{\mathbf{z}}, \mathbf{B}\mathbf{\Sigma}^{-2}\mathbf{B}^T). \quad (13)$$

  where $\overline{\mathbf{z}}$ is the mean vector and $\mathbf{\Sigma}$ is the eigenvalues that correspond to the principal subspace $\mathbf{B}$.
- $p_d(\mathbf{z}_t|\mathbf{A}_t)$ is the probability of a sample generated from the principal subspace spanned by the columns of $\mathbf{B}$. If we assume that the observation process is governed by an additive Gaussian model with a variance term $\epsilon\mathbf{I}$ then

$$\begin{aligned} p_d(\mathbf{z}_t|\mathbf{A}_t) &= \mathcal{N}(\mathbf{z}_t; \overline{\mathbf{z}}, \mathbf{B}\mathbf{B}^T + \epsilon\mathbf{I}) \\ \lim_{\epsilon\to 0} p_d(\mathbf{z}_t|\mathbf{A}_t) &\propto e^{-||(\mathbf{z}_t - \overline{\mathbf{z}}) - \mathbf{B}\mathbf{B}^T(\mathbf{z}_t - \overline{\mathbf{z}})||^2} \end{aligned} \quad (14)$$

Having defined models for $p(\mathbf{A}_t|\mathbf{A}_{t-1})$ and $p(\mathbf{z}_t|\mathbf{A}_t)$ the sequential inference model can be summarized in **Algorithm 4**.

IV. Results

The proposed tracker (which we coin FDR-PCA for the rest of the paper) is tested on several publicly available challenging video sequences which contain intrinsic and extrinsic changes to the tracked faces. The state-of-the-art IVT of Ross *et al.* [9] and its extension for IKPCA by Chin and Suter [10] act as comparison as they both form an appearance-based holistic tracker which classifies the foreground without additional background models. The initial position of the objects, the number of particles and the size of the eigenspaces are equivalent in all methods for each video sequence. Additionally, the results of another holistic tracker proposed by Zhou *et al.* [3] are included in the experiments.

For the proposed algorithm the parameter α, used by the kernel function (3) of the proposed FDR-PCA, should be a set *a-priori*. Different values were tested on a validation set of video sequences (different to the set of video sequences used for the experiments presented in this section) and for this validation set $\alpha = 0.7$ performed best, and therefore the parameter was fixed to this value. The variance of the Gaussian RBF kernel, used with the IKPCA algorithm, was selected in a similar manner.

A. Quantitative Evaluation

The Dudek video sequence[1] forms the data for the quantitative evaluation (fig. 3). In this sequence, each frame contains seven annotated positions of points which describe the true location and formation of the face. The points' initial position in the first frame are given and used to describe the initial transformation of the unit square for the holistic trackers. The trackers then estimate the transformation for subsequent frames, with which the new position of the points are calculated. The accuracy of the tracking in subsequence frames is then defined as the root mean square (RMS) error between the ground truth and the recognized points. Fig. 4 plots the RMS error for the whole Dudek video sequence for both the proposed and IVT methods.

The method of Zhou *et al.* [3] loses track after the occlusion between frame 100 and frame 120. IKPCA unsuccessfully estimates the motion in frame 288, after the filmed person rises from the chair in a quick movement. Only two methods, IVT and FDR-PCA, manage to follow the object for the whole length of the video. The mean RMS error of both methods are compared in table I. The proposed method performs most accurately.

TABLE I

Mean RMS Error on Dudek Video Sequence

Method	Mean RMS Error
IVT	7.45
FDR-PCA	6.79

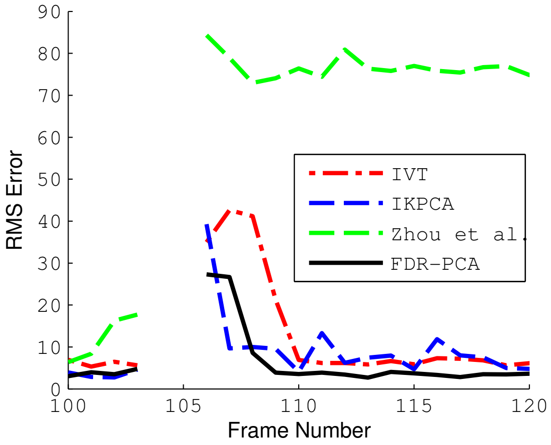


Fig. 1. RMS error of the different trackers before, during and after the occlusion between frame 100 and frame 120. (There is no value during the complete occlusion as ground-truth points are hidden.)

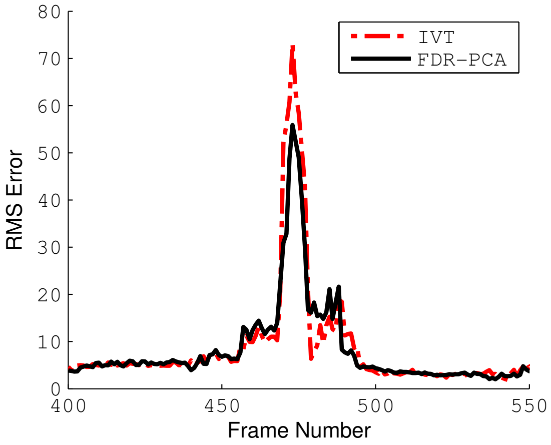


Fig. 2. RMS error of the proposed tracker (FDR-PCA) and IVT during pose variation (frame 450 to frame 470), and during motion blur (frame 480 to frame 500).

The RMS errors during the occlusion between frame 100 and frame 120 are compared (fig. 1). IKPCA performs competitively until the occlusion, however, RMS errors are higher thereafter for this method. IVT generally performs less accurately than IKPCA and FDR-PCA before the occlusion. The occlusion itself has little impact on this method, thus the algorithm continues on similar accuracy afterwards. The tracker proposed in this paper recovers most quickly from the occlusion: the effects of the occlusion are counteracted by the robustness of the scheme, and the overall displacement of the unit square is kept to a minimum. The accuracy of FDR-PCA during motion blur around frame 288 and frame 486 is slightly lower than IVT, but pose variation in frame 470 is better supported (fig. 2).

Finally, fig. 6 plots the the RMS error versus α. As can be seen, for a wide range of α values the algorithm performs rather well.

[1] The Dudek video sequence with annotations is available from: http://www.cs.toronto.edu/~dross/ivt/

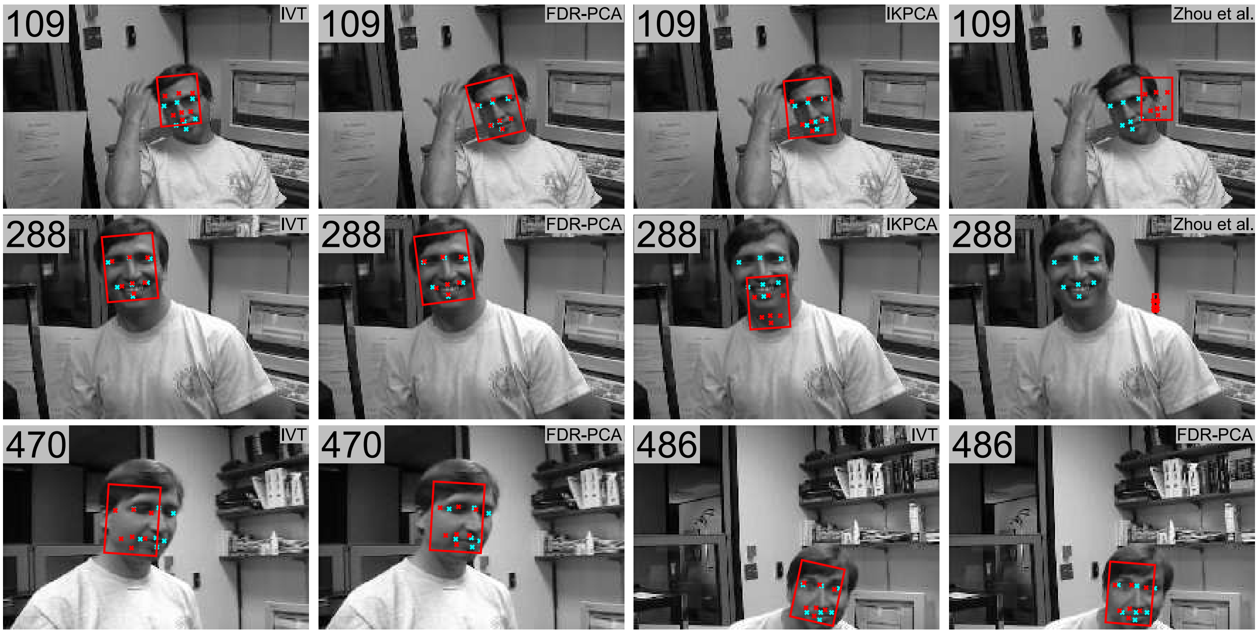


Fig. 3. Tracking results of the different schemes for the Dudek video sequence. The third and fourth column (first two rows) show Zhou *et al.* [3] and IKPCA – both trackers lose the object. IVT and the proposed tracker is shown in the first and second column (first two rows) respectively. The last row show two examples of some late frames for the proposed and the IVT tracker (the other tested trackes have already lost the face). The ground truth is indicated by cyan-colored points.

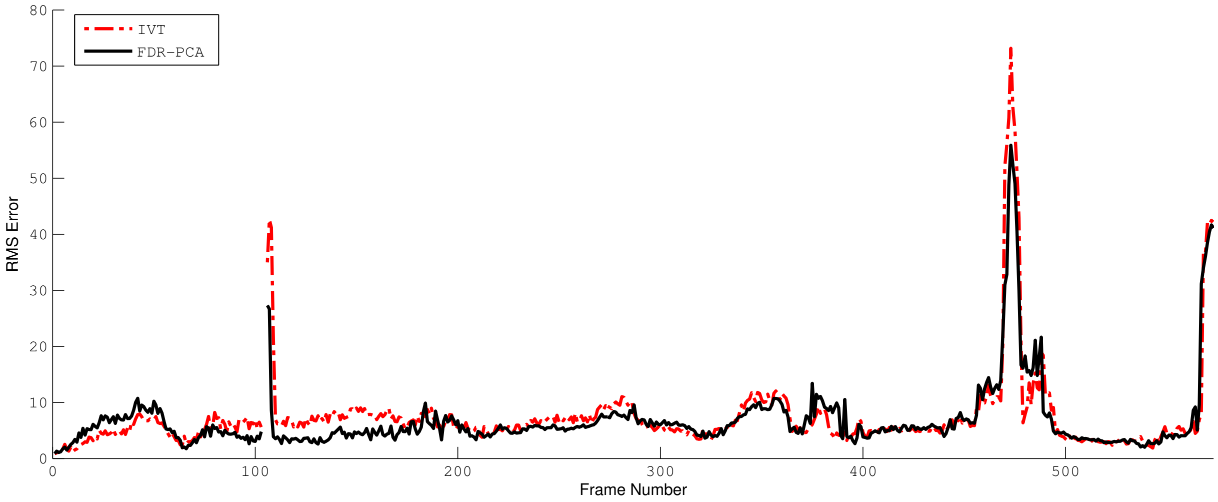


Fig. 4. RMS error of the proposed tracker and IVT for the whole sequence of Dudek.

B. *Qualitative Evaluation*

Three challenging video sequences[2] with challenging illuminations, occlussions and pose variations were used for the qualitative evaluation. Fig. 5 shows the results of the different trackers when the target object undergoes several pose changes and illumination alterations. IKPCA is the first method which loses the object in this sequence due to variations in the lighting condition. While the scheme of Zhou *et al.* copes with the change in frame 77, it fails after the extreme illumination changes in frame 172, just after the object moves from a bright into a dark area. IVT and the proposed tracker prove robust towards these type of changes, as both methods successfully track the objects until frame 329. The frames around frame 329 contain difficult prolonged pose changes, and therefore cause IVT to lose track in frame 329. The proposed FDR-PCA tracker successfully follows the face through all the frames of the video sequence until it eventually misclassifies the object's position in frame 330. Thus, for this video sequence, the proposed tracker outperforms other state-of-the-art trackers as it is more robust to illumination changes and pose variation.

Fig. 7.a shows the proposed tracker under variations in both, illumination and pose, and occlusion. In this sequence,

[2] The video sequences are available from: http://www.cs.toronto.edu/~dross/ivt/ and http://vision.ucsd.edu/~bbabenko/project_miltrack.shtml

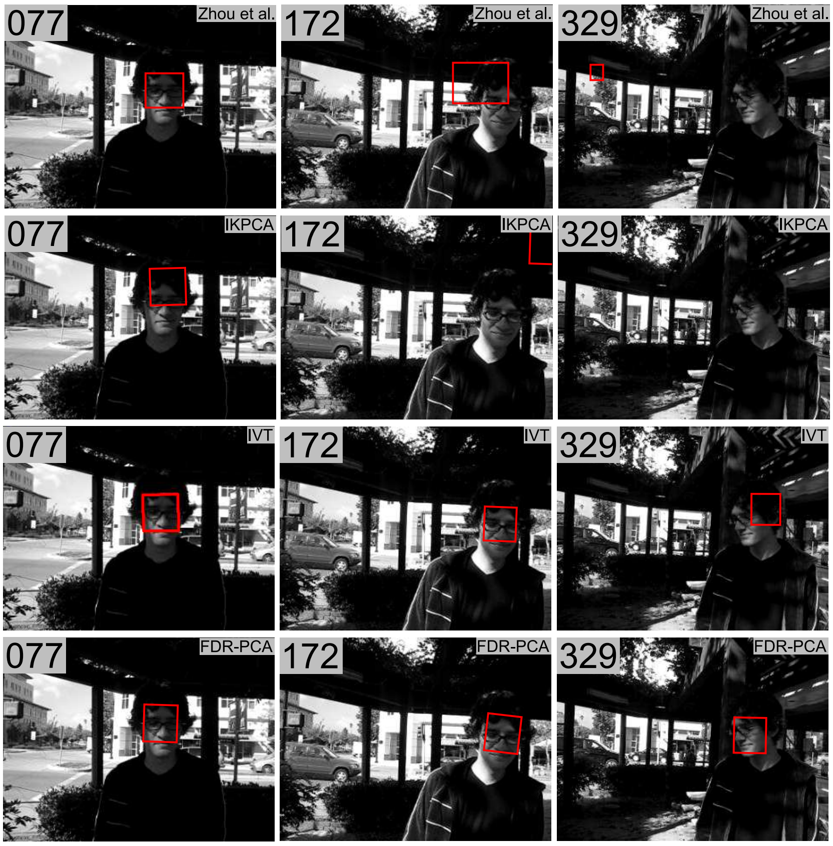


Fig. 5. Results of the different tracking schemes under extreme illumination changes and pose variation.

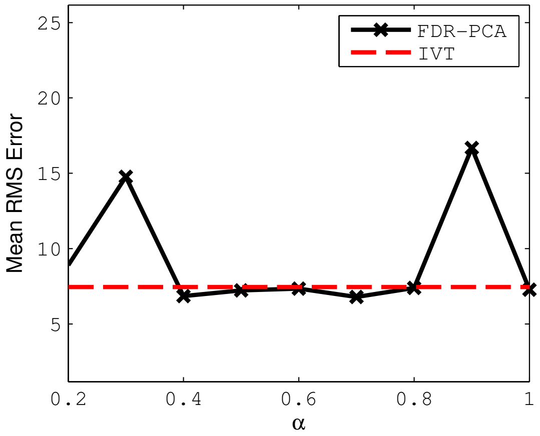


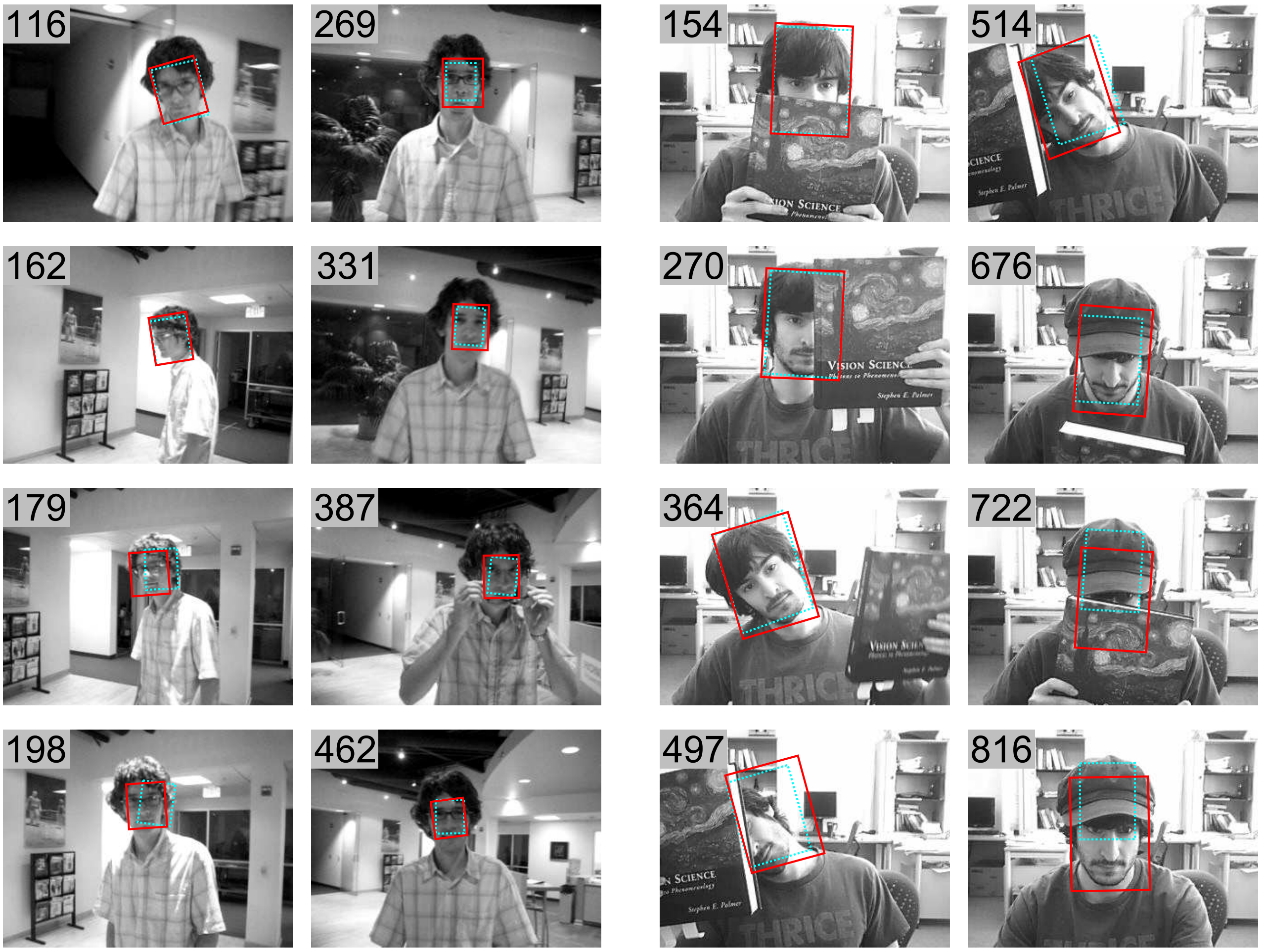
Fig. 6. Results of different α values for (3) when used for tracking the Dudek video sequence (section IV) with the proposed tracker.

the tracker successfully tracks the face throughout the complete sequence of frames. Even after the side-view of the face in frame 162, FDR-PCA recovers considerably better than IVT, and therefore the target is recognized correctly between frame 179 and frame 198. The occlusion in frame 331 and frame 387 is handled by both approaches.

The effect of occlusions on the proposed tracking scheme is presented by the video shown in fig. 7.b. The tracker quickly and successfully recovers from prolonged occlusions as in frame 497 and 722. In comparison to IVT, its performance is more robust for this sequence.

V. Conclusions and Future Work

We introduced a fast, direct and robust approach to incremental PCA for appearance-based visual tracking. Our results show that the proposed tracker is robust to illumination changes, some pose variations, intrinsic alterations and most prolonged occlusions. Our tracker outperforms existing holistic visual trackers in quantitative and qualitative evaluations. In contrast to IKPCA [10], the proposed scheme avoids the optimization required for finding the mean of the feature space with the implicit kernel function *via* pre-images, yet utilizes robust kernel PCA. Our tracker directly utilizes the incremental learning framework of IVT [9], and therefore not only is more robust but also equally fast. In future work, tracking may be improved by employing multiple adaptive expert appearance models for different views of the object. Within this framework, extreme changes in the object will initiate the generation of a new appearance model for this pose. Additionally, a more sophisticated particle generator for the particle filter which describes more than a simple condensation may be added. This may improve the efficiency as well as the accuracy of the proposed algorithm as fewer particles' likelihoods need to be calculated for better performance.



(a)

(b)

Fig. 7. The results of the proposed tracker (in solid red) compared to the IVT of Ross *et al.* [9] (in dotted cyan) on different video sequences.

VI. Acknowledgments

The work has been funded by the European Research Council under the ERC Starting Grant agreement no. ERC-2007-StG-203143 (MAHNOB). The work of Maja Pantic is also funded in part by the European Communitys 7th Framework Programme [FP7/20072013] under the grant agreement no 231287 (SSPNet). Stephan Liwicki is funded by an EPSRC DTA Studentship.

References

[1] M. Black and A. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," in *IJCV'98*, vol. 26, 1998, pp. 63 – 84.

[2] A. Jepson, D. Fleet, and T. El-Maraghi, "Robust Online Appearance Models for Visual Tracking," in *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, pp. 1296 – 1311.

[3] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters," in *IEEE Trans. Image Process.*, vol. 13, 2004, pp. 1491 – 1506.

[4] S. Avidan, "Support vector tracking," in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, 2004, pp. 1064 – 1072.

[5] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Int. joint Conf. on A. I.*, vol. 3, 1981, pp. 674 – 679.

[6] G. Hager and P. Belhumeur, "Efficient Region Tracking with Parametric Models of Geometry and Illumination," in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, 1998, p. 1025.

[7] S. Baker and I. Matthews, "Equivalence and Efficiency of Image Alignment Algorithms," in *CVPR'01*, vol. 1, 2001, pp. 1090 – 1097.

[8] I. Matthews and S. Baker, "Active Appearance Models Revisited," in *IJCV'04*, vol. 60, 2004, pp. 135 – 164.

[9] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental Learning for Robust Visual Tracking," in *IJCV'08*, vol. 77, 2008, pp. 125 – 141.

[10] T.-J. Chin and D. Suter, "Incremental Kernel Principal Component Analysis," in *IEEE Trans. Image Process.*, vol. 16, 2007, pp. 1662 – 1674.

[11] M. Isard and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," in *ECCV'96*, 1996, pp. 343 – 356.

[12] F. de la Torre and M. Black, "A Framework for Robust Subspace Learning," in *IJCV'03*, vol. 54, 2003, pp. 117 – 142.

[13] A. Fitch, A. Kadyrov, W. Christmas, and J. Kittler, "Fast Robust Correlation," in *IEEE Trans. Image Process.*, vol. 14, 2005, pp. 1063 – 1073.

[14] M. Turk and A. Pentland, "Eigenfaces for Recognition," in *Journal of Cognitive Neuroscience*, vol. 3, 1991, pp. 71 – 86.

[15] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," in *Neural computation*, vol. 10, 1998, pp. 1299 – 1319.

[16] M. Tipping and C. Bishop, "Probabilistic Principal Component Analysis," in *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, 1999, pp. 611 – 622.