TRACKING SPARSE SIGNAL SEQUENCES FROM NONLINEAR/NON-GAUSSIAN MEASUREMENTS AND APPLICATIONS IN ILLUMINATION-MOTION TRACKING

R. Sarkar, S. Das, N. Vaswani

ECE dept, Iowa State University, Ames IA

ABSTRACT

In this work, we develop algorithms for tracking time sequences of sparse spatial signals with slowly changing sparsity patterns, and other unknown states, from a sequence of nonlinear observations corrupted by (possibly) non-Gaussian noise. A key example of the above problem occurs in tracking moving objects across spatially varying illumination changes, where motion is the small dimensional state while the illumination image is the sparse spatial signal satisfying the slow-sparsity-pattern-change property.

***Index Terms*—** particle filtering, compressed sensing, tracking

1. INTRODUCTION

We study the problem of tracking (causally estimating) a time sequence of *sparse spatial signals with slowly changing sparsity patterns*, as well as *other unknown states*, from a sequence of *nonlinear observations corrupted by (possibly) non-Gaussian noise*. In many practical applications, the unknown state can be split into a small dimensional part and a spatial signal (large dimensional part). The spatial signal is often well modeled as being sparse in some domain. For a long sequence, its sparsity pattern can change over time, although the changes are usually slow. A key example of the above problem occurs in tracking moving objects across spatially varying illumination changes, e.g. persons walking under a tree (different lighting falling on different parts of the face due to the leaves blocking or not blocking the sunlight and this pattern changes with time as the leaves move) or in indoor sequences with variable lighting. In all these cases, one needs to explicitly track the motion (small dimensional part) as well as the illumination "image" (illumination at each pixel in the image), which is the spatial signal satisfying the slow-sparsity-pattern-change property [see Sec 4].

Related Work. In recent years, starting with the seminal papers of Candes, Romberg, Tao and of Donoho [1, 2] there has been a large amount of work on sparse signal recovery / compressive sensing (CS). The problem of recursively recovering a time sequence of sparse signals, with slowly changing sparsity patterns and signal values, from linear measurements has also been extensively studied [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15].

For tracking problems that need to causally estimate a time sequence of hidden states, X_t, from nonlinear and possibly non-Gaussian measurements, Y_t, the most common and efficient solution is to use a particle filter (PF). The PF uses sequential importance sampling [16] along with a resampling step [17] to obtain a sequential Monte Carlo estimate of the posterior distribution, $f_{X_t|Y_{1:t}}(x_t|y_{1:t})$, of the state X_t conditioned on all observations up to the current time, $Y_{1:t}$. In our problem, part of the state vector is a discrete spatial signal and hence very high dimensional. As a result, in this case, the original PF [17] will require too many particles for accurate tracking and hence becomes impractical to use. As explained in [18], the same is essentially true for most existing PF algorithms. Efficient PFs such as PF-Doucet[16], Gaussian PF [19], Gaussian sum filters or Gaussian sum PF [20] also cannot be used for the following reason. The first two implicitly assume that the posterior conditioned on the previous state, $f_{X_t|Y_t,X_{t-1}}(x_t|y_t, x_{t-1})$, is unimodal most of the time. The second two assume a linear, or at least, a unimodal observation model. In many problems, e.g. the illumination-motion tracking problem, neither assumption holds. For similar reasons, the extended Kalman filter (KF), the unscented KF, the interacting multiple mode filter or Gaussian mixture filters [21] also cannot be used.

Rao-Blackwellized PF (RB-PF) [22, 23], PF with posterior mode tracking (PF-MT) algorithm [24, 25, 26] and later works such as [27, 28, 29, 30] are possible solutions for large dimensional tracking problems. RB-PF requires that conditioned on the small dimensional state vector, the state space model be linear and Gaussian. This often does not hold. PF-MT relaxes this and only requires that, conditioned on the previous state and a small dimensional state vector, the posterior of the rest of the state vector be unimodal most of the time. However, neither RB-PF nor PF-MT exploits the sparsity of the spatial signal to be tracked. The same is true for [27, 28, 29, 30], as well as for works that introduce efficient resampling strategies [31, 32].

Contributions. (1) In this work, we exploit the fact that in most large dimensional tracking problems, at any given time, the large dimensional state vector is usually a spatial signal that is sparse in some dictionary/basis and at most times satisfies slow-sparsity-pattern-change, to propose a PF based tracking algorithm called Particle Filtered Modified-CS (nonlinear) or PaFiMoCS-nl. Unlike [3, 4, 6, 9, 15] which only solve the linear measurements' model case, PaFiMoCS-nl is designed for tracking problems with highly nonlinear, and possibly non-Gaussian, observation models that result in frequently multimodal observation likelihoods. Many visual tracking problems, e.g. tracking moving objects across spatially varying illumination change [18, 33], fit into this category. We introduced PaFiMoCS to approximately compute the causal minimum mean squared error estimate of sparse signal sequences from linear Gaussian measurements in [15]. In this work, we design PaFiMoCS-nl for recovering sparse signal sequences from nonlinear/non-Gaussian measurements and we develop an improved PaFiMoCS-nl algorithm for very large sized problems.

(2) Our experiments with the dictionary of Legendre polynomials (henceforth referred to as the Legendre dictionary) are the first to demonstrate that for many videos with significant illumination variations, the illumination image is approximately sparse in this dictionary. Also, its sparsity pattern includes many of the higher order Legendre polynomials, and may not include all the lower order ones (as was assumed in earlier works [34, 33, 18]). Moreover, over time, the sparsity pattern usually changes quite slowly [see Sec 4)]. We have explained in the long version [35] why we pick this dictionary

This work was partly supported by NSF grant IIS-1117509.

and not something simpler like Fourier.

Notation. The probability density function (PDF) or probability mass function (PMF) of a random variable (r.v.) Y is denoted by $f_Y(y)$ while the conditional PDF of of r.v. Y given another r.v. X is denoted by $f_{Y|X}(y|x)$. We use $\mathbb{E}[Y]$ for expectation. The subscript t denotes the discrete time index. The notation $Z_t \overset{\text{i.i.d.}}{\sim} f(z)$ means that the sequence of r.v.'s $Z_1, Z_2, \dots Z_t, \dots$ are independent and identically distributed (i.i.d.) with PDF or PMF $f(z)$. The notation $S \sim \text{Ber}(T, p)$ means that the set S contains any element of the set T with probability p (and does not contain it with probability $1-p$) independent of all other elements of T. The notation $\mathcal{N}(a; \mu, \Sigma)$ denotes the value of a Gaussian PDF with mean μ and covariance matrix Σ computed at a. The notation $X \sim \mathcal{N}(\mu, \Sigma)$ means that the r.v. X is Gaussian distributed with mean μ and covariance matrix Σ.

For a set T, $T^c := \{i : i \notin T\}$; $\cup$ and $\cap$ denote set-union and set-intersection respectively; set difference, $T_1 \setminus T_2 := T_1 \cap T_2^c$; and $|T|$ denotes its cardinality. For a vector b, $\|b\|_k$ denotes its ℓ_k norm and $(b)_T$ denotes the sub-vector containing the elements of b with indices in T. Also, $(.)^\top$ denotes transpose. The notation vec(.) converts a matrix into a vector by cascading the rows and $\odot$ denotes the Hadamard product.

2. THE PROBLEM AND STATE TRANSITION MODELS

The goal is to recursively recover a time sequence of states, X_t, from noise-corrupted and nonlinear measurements, Y_t, when X_t can be split into two parts,

1. a small dimensional part, U_t, and
2. a large dimensional part, L_t, with the following property: L_t is a spatial signal, that is sparse in some known dictionary, and its sparsity pattern usually changes slowly over time.

Mathematically, this means the following. The state X_t can be split as $X_t = \begin{bmatrix} U_t \\ L_t \end{bmatrix}$ where $(U_t)_{n_u \times 1}$ is a small dimensional state vector and $(L_t)_{n_l \times 1}$, with $n_l \gg n_u$, is a discrete spatial signal that is sparse in some known dictionary, Φ, i.e.

$$L_t = \Phi \Lambda_t, \tag{1}$$

where $(\Lambda_t)_{n_\lambda \times 1}$ is a sparse vector with support set T_t, i.e.

$$T_t := \text{support}(\Lambda_t) = \{j : (\Lambda_t)_j \neq 0\}. \tag{2}$$

The $n_l \times n_\lambda$ dictionary matrix Φ can be tall, square or fat.

Notice that, if T_t and $(\Lambda_t)_{T_t}$ are known, then L_t is known. *Thus, one can as well let the state vector be $X_t = [U_t^\top, T_t^\top, {(\Lambda_t)_{T_t}}^\top]^\top$ or for simplicity, just $X_t = [U_t^\top, T_t^\top, \Lambda_t^\top]^\top$. We will use this definition of the state vector in the rest of this paper.*

We assume that the observation vector, $(Y_t)_{m \times 1}$, satisfies

$$g(Y_t, U_t, L_t) = Z_t, \; Z_t \overset{\text{i.i.d.}}{\sim} f_Z(z) \tag{3}$$

where Z_t is the observation noise. For the above model, the observation likelihood, $\text{OL}(U_t, L_t)$, can be written as

$$\text{OL}(U_t, L_t) := f_{Y_t|U_t, L_t}(Y_t|U_t, L_t) = f_Z(g(Y_t, U_t, L_t)) \tag{4}$$

2.1. State Transition Models

In the absence of specific model information, one can adopt the following simple state transition models. For T_t, we assume that

$$T_t = T_{t-1} \cup A_t \setminus R_t, \text{ where}$$
$$A_t \overset{\text{i.i.d.}}{\sim} \text{Ber}(T_{t-1}^c, p_a), \;\; R_t \overset{\text{i.i.d.}}{\sim} \text{Ber}(T_{t-1}, p_r) \tag{5}$$

Here A_t denotes the set added to the support at time t, while R_t denotes the set that is removed from the support at time t. Each of them is i.i.d. over time. Conditioned on T_{t-1}, A_t and R_t are independent of each other. In most applications, it is valid to assume that the expected support size remains constant, i.e. $\mathbb{E}[|T_t|] = \mathbb{E}[|T_{t-1}|] = s$. This is ensured by setting $p_r = (n_\lambda - s)p_a/s$ so that $\mathbb{E}[|A_t|] = (n_\lambda - s)p_a = \mathbb{E}[|R_t|] = sp_r$. Also, *slow support change* means that $\mathbb{E}[|R_t|] = \mathbb{E}[|A_t|]$ is small compared to $\mathbb{E}[|T_t|] = s$. This is ensured by picking p_a to be small compared with $s/(n_\lambda - s)$.

In the absence any of other model information, we assume the following linear Gaussian random walk model on $(\Lambda_t)_{T_t}$ and U_t:

$$(\Lambda_t)_{T_t} = (\Lambda_{t-1})_{T_t} + (\nu_{l,t})_{T_t}, \; (\nu_{l,t})_{T_t} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_l^2 I)$$
$$(\Lambda_t)_{T_t^c} = 0 \tag{6}$$

$$U_t = U_{t-1} + \nu_{u,t}, \; \nu_{u,t} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \Sigma_u) \tag{7}$$

The state transition prior corresponding to the above state models can be written as: $\text{STP}(X_t^i; X_{t-1}^i) := f_{X_t|X_{t-1}}(X_t^i|X_{t-1}^i) = \text{STP}(T_t^i; T_{t-1}^i) \, \text{STP}(\Lambda_t^i; \Lambda_{t-1}^i, T_t^i) \, \text{STP}(U_t^i; U_{t-1}^i)$ where

$$\text{STP}(T_t^i; T_{t-1}^i) = p_a^{|T_t^i \setminus T_{t-1}^i|} (1-p_a)^{n_l - |T_{t-1}^i| - |T_t^i \setminus T_{t-1}^i|} \times$$
$$p_r^{|T_{t-1}^i \setminus T_t^i|} (1-p_r)^{|T_{t-1}^i| - |T_{t-1}^i \setminus T_t^i|}, \tag{8}$$

$$\text{STP}(\Lambda_t^i; \Lambda_{t-1}^i, T_t^i) = \mathcal{N}((\Lambda_t^i)_{T_t^i}; (\Lambda_{t-1}^i)_{T_t^i}, \sigma_l^2 I), \tag{9}$$

$$\text{STP}(U_t^i; U_{t-1}^i) = \mathcal{N}(U_t^i; U_{t-1}^i, \Sigma_u). \tag{10}$$

3. PARTICLE FILTERED MODIFIED-CS (NONLINEAR)

Particle Filtered Modified-CS (PaFiMoCS) is inspired by RB-PF [23, 22] and PF-MT [24]. PF-MT [24] splits the state vector X_t into $X_t = [X_{t,s}^\top, X_{t,r}^\top]^\top$ where $X_{t,s}$ denotes the coefficients of a small dimensional state vector, which can change significantly over time, while $X_{t,r}$ refers to the rest of the states (large dimensional) which usually change much more slowly over time. PF-MT importance samples only on $X_{t,s}$, while replacing importance sampling by deterministic posterior Mode Tracking (MT) for $X_{t,r}$ and thus significantly reducing the importance sampling dimension. PF-MT can be applied to our problem if we replace (5) and (6) by (6) with $T_t = [1, 2 \dots n_\lambda]$, i.e. we do not use the sparsity of Λ_t, and we let $X_{t,s} = U_t$ and $X_{t,r} = \Lambda_t$. However, since PF-MT does not exploit the sparsity or slow sparsity pattern change of Λ_t, it is very likely that it will result in a dense solution for Λ_t, i.e. the energy will get distributed among all components of Λ_t. This is a problem for applications where Λ_t is indeed well approximated by a sparse vector with changing sparsity patterns. An alternative could be to assume (6) on a selected fixed subset of Λ_t, i.e. fix $T_t = T_0$. For example, if Φ is the Fourier basis or a Legendre dictionary, one would pick the first few components as the set T_0. This was done in [18] for illumination. This approach works if most energy of L_t does indeed lie in the lower frequency (or lower order Legendre) components, but fails if there are different types of high-frequency (higher order Legendre) spatial variations in L_t over time[1]. For many of the video sequences we experimented with for motion tracking across illumination change, this latter assumption was true [see Sec 4], and as a result PF-MT implemented this way also failed.

3.1. PaFiMoCS-nl: Particle Filtered Modified-CS (nonlinear)

To address the above limitation, one can utilize the sparsity and slow sparsity pattern change of the large dimensional state vector, L_t, in

[1] Higher order Legendre polynomials roughly correspond to higher frequency spatial variations of intensity.

Algorithm 1 PaFiMoCS-nl: PF Modified-CS (nonlinear)

Input: Y_t, Output: $U_t^i, T_t^i, \Lambda_t^i, w_t^i$. For all $t \geq 0$ do

1. For each particle i: sample $U_t^i \sim \mathcal{N}(0, \Sigma_u)$
2. For each particle i: set $T_t^i = T_{t-1}^i \cup A_t^i \setminus R_t^i$ where $A_t^i \sim \text{Ber}((T_{t-1}^i)^c, p_a)$ and $R_t^i \sim \text{Ber}(T_{t-1}^i, p_r)$.
3. For each particle i: mode track Λ_t with imposing slow sparsity pattern change, i.e. compute Λ_t^i as the solution of (11) with OL(.) as defined in (4).
4. For each particle i: update T_t^i as $T_t^i := \{j : |(\Lambda_t^i)_j| > \alpha\}$.
5. For each particle i: compute the weights as follows
$$w_t^i \propto w_{t-1}^i \, \text{OL}(U_t^i, \Phi\Lambda_t^i) \, \text{STP}(\Lambda_t^i; \Lambda_{t-1}^i, T_t^i)$$
where OL(.) is defined in (4) and $\text{STP}(\Lambda_t^i; \Lambda_{t-1}^i, T_t^i)$ is defined in (10). Resample and reset weights to $1/n_{pf}$.

a PF-MT type framework as follows. We let $X_{t,s} = [U_t, T_t]$ and $X_{t,r} = \Lambda_t$. In the importance sampling step, we sample U_t^i and T_t^i from their state transition priors given in Sec 2.1. Motivated by modified-CS [6], we add a term of the form $\|\Lambda_{T^c}\|_1$ with $T = T_t^i$ in the mode tracking cost function, i.e. it computes Λ_t^i by solving

$$\min_{\Lambda} C(\Lambda), \ C(\Lambda) := -\log \text{OL}(U_t^i, \Phi\Lambda) + \beta \frac{\|(\Lambda - \Lambda_{t-1}^i)_{T_t^i}\|_2^2}{2\sigma_l^2} + \gamma \|\Lambda_{(T_t^i)^c}\|_1 \quad (11)$$

where OL(.) is defined in (4). Solving (11) is a tractable approximation to trying to find the vector Λ_t^i that is sparsest outside the set T_t^i (i.e. the vector with the smallest number of new support additions to T_t^i) among all vectors Λ that satisfy the observation model constraint (often referred to as the data constraint) and are "close enough" to the previous estimate, $(\Lambda_{t-1}^i)_{T_t^i}$. Thus solving (11) ensures that the support of the solution, Λ_t^i, does not change too much w.r.t. the predicted support particle T_t^i. The larger the value of γ, the smaller will be the support change. A second change that we have w.r.t. the original PF-MT idea is that we threshold on Λ_t^i in order to get an updated estimate of the support T_t. We summarize the resulting algorithm in Algorithm 1. We refer to it as *Particle Filtered Modified-CS-nonlinear (PaFiMoCS-nl)*.

3.2. PaFiMoCS-nl-slow-support-change: PaFiMoCS-nl for very large problems with slow support changes

For certain problems with very large sized spatial signals, L_t, the support size of its sparse coefficients vector, Λ_t, can also be very large. In these situations, if we keep T_t as part of the importance sampling state $X_{t,s}$, it will require a large number of particles, thus making the algorithm impractical. However, *if the support changes slowly enough*, then we can include T_t as part of $X_{t,r}$, i.e. we let $X_{t,s} = U_t$ and $X_{t,r} = [T_t, \Lambda_t]$. With this, the mode tracking step would ideally have to compute Λ_t^i, A_t^i and R_t^i by solving $\min_{\Lambda,A,R} C(\Lambda, A, R)$, where

$$C(\Lambda, A, R) := -\log \text{OL}(U_t^i, \Phi\Lambda) + \beta \frac{\|(\Lambda - \Lambda_{t-1}^i)_{T_{t-1}^i}\|_2^2}{2\sigma_l^2} - |A| \log \frac{p_a}{1-p_a} - |R| \log \frac{p_r}{1-p_r} \quad (12)$$

and setting $T_t^i = T_{t-1}^i \cup A_t^i \setminus R_t^i$. But the above minimization will require a brute force approach of checking all possible sets A and R and will thus have complexity that is exponential in the support change size. Thus, it cannot be solved in any reasonable time. However, we can instead compute Λ_t^i by solving

$$\min_{\Lambda} C(\Lambda), \ C(\Lambda) := -\log \text{OL}(U_t^i, \Phi\Lambda) + \beta \frac{\|(\Lambda - \Lambda_{t-1}^i)_{T_{t-1}^i}\|_2^2}{2\sigma_l^2} + \gamma \|\Lambda_{(T_{t-1}^i)^c}\|_1 \quad (13)$$

and then threshold on Λ_t^i to get the current support particle T_t^i. Since $p_a < 0.5$ and $p_r < 0.5$, the last two terms of (12) are increasing functions of $|A|$ and $|R|$. If the last term of (12) were ignored, doing the above can be interpreted as its convex relaxation: it helps to find the vector Λ_t^i with the smallest number of support additions to T_{t-1}^i, i.e. the smallest $|A_t^i|$, while also keeping the first two terms small.

Since the support set T_t is now a part of $X_{t,r}$, we also need to include a term proportional to its state transition prior in the weighting step, i.e. we need to also multiply by (8) in the weighting step. We summarize the resulting algorithm in Algorithm 2. We refer to it as *PaFiMoCS-nl-slow-support-change*.

Algorithm 2 PaFiMoCS-nl-slow-support-change

Input: Y_t, Output: $U_t^i, T_t^i, \Lambda_t^i, w_t^i$. For all $t \geq 0$ do

1. For each particle i: sample $U_t^i \sim \mathcal{N}(0, \Sigma_u)$
2. For each particle i: compute Λ_t^i as the solution of (13) with OL(.) as defined in (4). Compute $T_t^i := \{j : |(\Lambda_t^i)_j| > \alpha\}$.
3. For each particle i: compute the weights as follows
$$w_t^i \propto w_{t-1}^i \text{OL}(U_t^i, \Phi\Lambda_t^i) \, \text{STP}(\Lambda_t^i; \Lambda_{t-1}^i, T_t^i) \, \text{STP}(T_t^i; T_{t-1}^i)$$
where OL(.) is defined in (4), $\text{STP}(\Lambda_t^i; \Lambda_{t-1}^i, T_t^i)$ is defined in (10) and $\text{STP}(T_t^i; T_{t-1}^i)$ is defined in (8). Resample and reset weights to $1/n_{pf}$. Increment t and go to step 1.

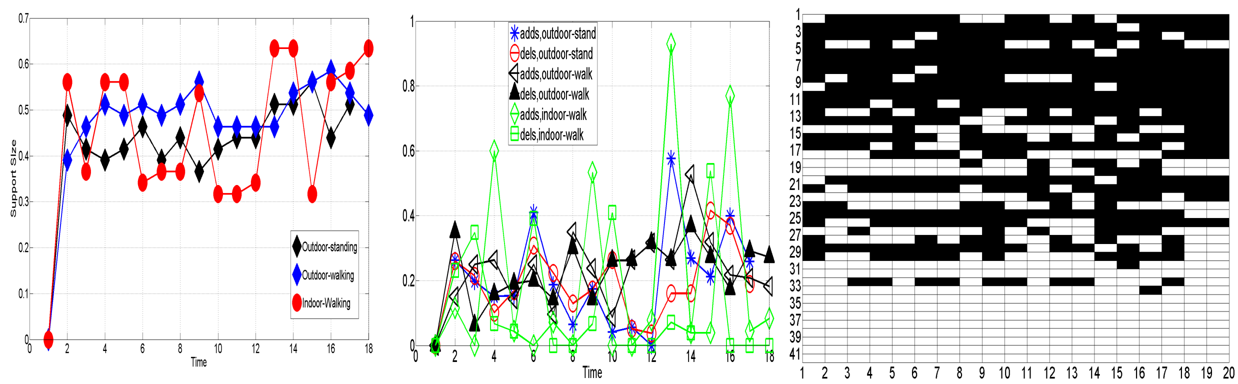
4. APPLICATION: ILLUMINATION-MOTION TRACKING

We show how visual tracking across spatially varying illumination change is an example of the general problem studied here. The state in this case consists of the $n_u \times 1$ motion state, U_t, which is the small dimensional part, and the $n_l \times 1$ illumination "image" (written as 1-D vector), L_t. We use a template-based tracking framework, similar to the one in [18, 33], with a simple three-dimensional motion model, that only models x-y translation and scale, i.e. $U_t = [u_t^x, u_t^y, s_t]^\top$ where s_t refers to scale and u_t^x and u_t^y refer to x and y translation. Thus $n_u = 3$. The illumination image, L_t is represented in the Legendre dictionary. Thus, our final state vector is $X_t = [U_t^\top, T_t^\top, \Lambda_t^\top]^\top$ where U_t is the $n_u \times 1$ motion state; Λ_t is the $n_\lambda \times 1$ Legendre coefficients' vector of illumination; and T_t is the support set of Λ_t. The observation model is taken from [18].

The initial template is denoted by I_0. We use ROI(U_t) to denote the region-of-interest (ROI) in the current image. It is obtained by scaling and translating the pixel locations of the original template I_0. The pixels outside the ROI, i.e. those in ROI$(U_t)^c$, are assumed to be due to clutter. We model them as being i.i.d. uniformly distributed between zero and 255. Thus, the current image Y_t satisfies

$$Y_t(\text{ROI}(U_t)) = \text{vec}(I_0) + \Phi\Lambda_t + Z_t, \ Z_t \sim \mathcal{N}(0, \sigma_o^2 I),$$
$$Y_t(\text{ROI}(U_t)^c) = Z_{t,c}, \ (Z_{t,c})_i \overset{\text{i.i.d.}}{\sim} \text{Unif}(0, 255) \quad (14)$$

where "i.i.d." means i.i.d. over i and t, ROI(U_t) is defined in [35,



(a) support size (b) support change size (c) support membership for outdoor-walking seq

Fig. 1. (a) The size of the 99%-support of the Legendre coefficients' vector, Λ_t, as a ratio of the length of Λ_t, i.e. $\frac{|T_t|}{n_\lambda}$ is plotted against time. (b) The number of additions, $\frac{|T_t \setminus T_{t-1}|}{|T_t|}$, and removals, $\frac{|T_{t-1} \setminus T_t|}{|T_t|}$, from this support set as a ratio of the support size are plotted against time. (c) We show the entries of the support set T_t at various times t. We shade in black the squares corresponding to indices that are contained in T_t, while leaving blank the indices that are not in T_t. The x-axis is time and the y-axis is the indices from 1 to $n_\lambda = 41$.

Sec IV-A, equation (20)],

$$\Phi \triangleq [\text{vec}(I_0 \odot P_0), ..., \text{vec}(I_0 \odot P_{2d})], \text{ where}$$

$$P_k(i,j) = \begin{cases} 1 \text{ if } k = 0 \\ p_{\frac{k+1}{2}}(i) \text{ if } k = 1,3,5,\ldots(2d-1) \\ p_{\frac{k}{2}}(j) \text{ if } k = 2,4,6,\ldots 2d \end{cases} \quad (15)$$

and $p_k(.)$ is the Legendre polynomial of k^{th} order. Thus, Φ is an $n_l \times n_\lambda$ matrix with $n_\lambda = 2d+1$. In our experiments, we used $d = 20$, so that $n_\lambda = 41$. As explained in [18], the above model is frequently multimodal due to background clutter.

In the absence of any extra information about the motion, we assume that U_t satisfies (7) with Σ_u being a diagonal matrix. As we show below, the Legendre coefficients vector for illumination, Λ_t, is an approximately sparse vector with support that usually changes slowly over time. Hence, the models given in Sec 2.1 apply for Λ_t as well, i.e. its support, T_t, satisfies (5) and Λ_t satisfies (6).

We used the video sequences shown in [35, Fig 1] to study sparsity pattern change of the illumination images over time. The first video (outdoor-standing) is of a person standing under a tree on a very windy day. As the leaves move, the illumination pattern on her face changes with time. The second video (outdoor-walking) is of a person walking under a tree with variable amounts of light falling on various parts of her face as she moves under the moving leaves. The third (indoor-walking) has a person walking along a corridor across a window. In a 20-frame sub-sequence of each video, we hand-marked a rectangle around the person's face in each frame. The details of how we computed L_t, Λ_t and its approximate support set T_t are explained in [35, Sec IV-C]. As can be seen from Fig 1(a), for all videos, the support size is between 30-60% of the length of Λ_t, n_λ. Thus, Λ_t is indeed approximately sparse. Also, as can be seen from Fig 1(b), except at a few time instants, the number of support changes is usually under 35% of the support size. For the indoor-walking sequence, at certain times (when the person moves towards the window from a darker region of the corridor or vice versa), this number is larger. From the support membership plot of Fig 1(c), we can see that the support set does indeed contain many of the higher order Legendre polynomials. Polynomials up to the 16^{th} order are present. Similar trends were seen also for the other two videos [35, Fig 3]. This explains why PF-MT run with only a 7-dimensional Λ_t ($d = 3$) as in [18] fails to track these sequences (see [35]).

5. RESULTS ON SIMULATED VIDEO SEQUENCES

We show only results on simulated videos here. Results on real videos are shown in [35, Sec V-A]. Starting with a face template, 50 video sequences of a moving target with spatially varying illumination change and background clutter were generated using the state transition models of Sec 2.1 and the observation model given in (14). Details in [35, Sec V-B]. Each sequence was tracked using PaFiMoCS (Algorithm 1) and PaFiMoCS-slow-support-change (Algorithm 2) with $d = 20$ as well as using PF-MT, Auxiliary-PF [36] and PF-Gordon [17], using both $d = 3$ and $d = 20$. All algorithms used 100 particles. In Fig 2, we plot $NMSE(t) := \frac{\mathbb{E}[\|U_t - \hat{U}_t\|_2^2 + \|\Lambda_t - \hat{\Lambda}_t\|_2^2]}{\mathbb{E}[\|U_t\|_2^2 + \|\Lambda_t\|_2^2]}$ against time. Here $\hat{U}_t$ and $\hat{\Lambda}_t$ are the weighted means of the particles of U_t and Λ_t respectively. $\mathbb{E}[.]$ denotes the Monte Carlo average.

As can be seen, PaFiMoCS remains in track with stable and small error throughout. PaFiMoCS-slow-support-change (PaFiMoCS-SSC) also remains in track, but its errors are slightly larger because occasionally the number of support changes was large. PF-MT-3 (the algorithm used in [18]) loses track because it assumes that only the first 7 Legendre polynomials are sufficient to represent the illumination image. However, we know from our simulation that the support of the illumination vector is equally likely to contain any element from $[1, 2, \ldots 41]$ (not just the first 7). PF-MT-20 loses track because it assumes that Λ_t is a dense vector, i.e. all of its 41 components are part of the support at all times. Aux-PF-20, Aux-PF-3, PF-Gordon-20 and PF-Gordon-3 lose track due to similar reasons and because 100 particles is too less for these algorithms.

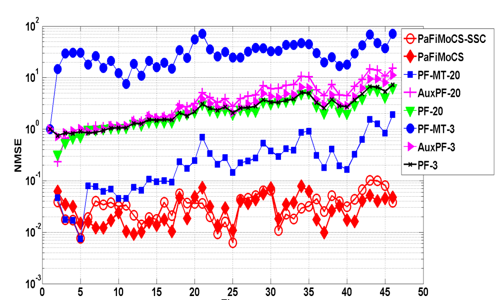


Fig. 2. Normalized mean squared error (NMSE) plot

6. REFERENCES

[1] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Info. Th.*, vol. 52(2), pp. 489–509, February 2006.

[2] D. Donoho, "Compressed sensing," *IEEE Trans. on Information Theory*, vol. 52(4), pp. 1289–1306, April 2006.

[3] N. Vaswani, "Kalman filtered compressed sensing," in *IEEE Intl. Conf. Image Proc. (ICIP)*, 2008.

[4] N. Vaswani, "LS-CS-residual (LS-CS): Compressive Sensing on Least Squares residual," *IEEE Trans. Sig. Proc.*, vol. 58(8), pp. 4108–4120, August 2010.

[5] A. Carmi, P. Gurfil, and D. Kanevsky, "Methods for sparse signal recovery using kalman filtering with embedded pseudo-measurement norms and quasi-norms," *IEEE Trans. Sig. Proc.*, pp. 2405–2409, April 2010.

[6] N. Vaswani and W. Lu, "Modified-cs: Modifying compressive sensing for problems with partially known support," *IEEE Trans. Sig. Proc.*, vol. 58(9), pp. 4595–4607, September 2010.

[7] A. Khajehnejad, W. Xu, A. Avestimehr, and B. Hassibi, "Weighted ℓ_1 minimization for sparse recovery with prior information," *IEEE Trans. Sig. Proc.*, 2011.

[8] M. Stojnic, "Block-length dependent thresholds for ℓ_2/ℓ_1-optimization in block-sparse compressed sensing," in *ICASSP*, 2010.

[9] W. Lu and N. Vaswani, "Regularized modified bpdn for noisy sparse reconstruction with partial erroneous support and signal value knowledge," *IEEE Trans. Sig. Proc.*, January 2012.

[10] D. Sejdinovic, C. Andrieu, and R. Piechocki, "Bayesian sequential compressed sensing in sparse dynamical systems," in *Allerton Conf. Communication, Control, and Computing*, 2010.

[11] J. Ziniel, L. C. Potter, and P. Schniter, "Tracking and smoothing of time-varying sparse signals via approximate belief propagation," in *Asilomar Conf. on Sig. Sys. Comp.*, 2010.

[12] Z. Zhang and B. D. Rao, "Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning," *IEEE J. Sel. Topics Sig. Proc., Special Issue on Adaptive Sparse Representation of Data and Applications in Signal and Image Processing*, vol. 5, no. 5, pp. 912–926, Sept 2011.

[13] A. Charles, M. S. Asif, J. Romberg, and C. Rozell, "Sparsity penalties in dynamical system estimation," in *Conf. Info. Sciences and Systems*, 2011.

[14] A. Y. Carmi and L. Mihailova, "Unscented compressed sensing," in *IEEE Intl. Conf. Acoustics, Speech, Sig. Proc. (ICASSP)*, 2012.

[15] S. Das and N. Vaswani, "Particle filtered modified compressive sensing (pafimocs) for tracking signal sequences," in *Asilomar*, 2010.

[16] A. Doucet, "On sequential monte carlo sampling methods for bayesian filtering," in *Technical Report CUED/F-INFENG/TR. 310, Cambridge University Department of Engineering*, 1998.

[17] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/nongaussian bayesian state estimation," *IEE Proceedings-F (Radar and Signal Processing)*, pp. 140(2):107–113, 1993.

[18] S. Das, A. Kale, and N. Vaswani, "Particle filter with mode tracker (pf-mt) for visual tracking across illumination changes," *IEEE Trans. Image Proc.*, April 2012.

[19] J. H. Kotecha and P. M. Djuric, "Gaussian particle filtering," *IEEE Trans. Sig. Proc.*, pp. 2592–2601, Oct 2003.

[20] J. H. Kotecha and P. M. Djuric, "Gaussian sum particle filtering," *IEEE Trans. Sig. Proc.*, pp. 2602–2612, Oct 2003.

[21] A. Doucet, N. deFreitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, Springer, 2001.

[22] T. Schn, F. Gustafsson, and P. Nordlund, "Marginalized particle filters for nonlinear state-space models," *IEEE Trans. Sig. Proc.*, 2005.

[23] R. Chen and J.S. Liu, "Mixture kalman filters," *Journal of the Royal Statistical Society*, vol. 62(3), pp. 493–508, 2000.

[24] N. Vaswani, "Particle filtering for large dimensional state spaces with multimodal observation likelihoods," *IEEE Trans. Sig. Proc.*, pp. 4583–4597, October 2008.

[25] N. Vaswani, A. Yezzi, Y. Rathi, and A. Tannenbaum, "Particle filters for infinite (or large) dimensional state spaces-part 1," in *IEEE Intl. Conf. Acoustics, Speech, Sig. Proc. (ICASSP)*, 2006.

[26] N. Vaswani, "Particle filters for infinite (or large) dimensional state spaces-part 2," in *IEEE Intl. Conf. Acoustics, Speech, Sig. Proc. (ICASSP)*, 2006.

[27] W. Y. Chang, C. S. Chen, and Y. D. Jian, "Visual tracking in high-dimensional state space by appearance-guided particle filtering," *Trans. Img. Proc.*, vol. 17, no. 7, pp. 1154–1167, July 2008.

[28] L. Mihaylova and A. Carmi, "Particle algorithms for filtering in high dimensional state spaces: A case study in group object tracking," in *ICASSP*, 2011, pp. 5932–5935.

[29] L. Mihaylova, A. Hegyi, A. Gning, and R. K. Boel, "Parallelized particle and gaussian sum particle filters for large-scale freeway traffic systems," *IEEE Trans. on Intell. Transport. Sys.*, vol. 13, no. 1, pp. 36–48, 2012.

[30] E. Besada-Portas, S. M. Plis, J. M. de la Cruz, and T. Lane, "Adaptive parallel/serial sampling mechanisms for particle filtering in dynamic bayesian networks," in *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part I*, Berlin, Heidelberg, 2010, pp. 119–134, Springer-Verlag.

[31] B. Balasingam, M. Bolic, P. M. Djuric, and J. Míguez, "Efficient distributed resampling for particle filters," in *ICASSP*, 2011.

[32] N. Zheng, Y. Pan, and X. Yan, "Hierarchical resampling architecture for distributed particle filters," in *ICASSP*, 2012.

[33] A. Kale and C. Jaynes, "A joint illumination and shape model for visual tracking," in *IEEE Conf. on Comp. Vis. Pat. Rec. (CVPR)*, 2006, pp. 602–609.

[34] Y. Weiss, "Deriving intrinsic images from image sequences," in *IEEE Intl. Conf. on Computer Vision (ICCV)*, 2001.

[35] R. Sarkar, S. Das, and N. Vaswani, "Pafimocs: Particle filtered modified-cs and applications in visual tracking across illumination change," *arXiv:1301.1374 [cs.CV]*, 2013.

[36] M. Pitt and N. Shephard, "Filtering via simulation: auxiliary particle filters," *J. Amer. Stat. Assoc*, vol. 94, pp. 590599, 1999.