ONLINE DATA-DRIVEN DYNAMIC IMAGE RESTORATION USING DINO-KAT MODELS

Brian E. Moore and Saiprasad Ravishankar

Department of EECS, University of Michigan, Ann Arbor, MI, USA

ABSTRACT

Sparsity-based techniques have been popular for reconstructing images and videos from limited or corrupted measurements. Methods such as dictionary or transform learning have been demonstrated to be useful in applications such as denoising, inpainting, and medical image reconstruction. In this work, we propose a new framework for online or sequential adaptive reconstruction of dynamic image sequences from linear (typically undersampled) measurements. In particular, the spatiotemporal patches of the underlying dynamic image sequence are assumed to be sparse in a DIctioNary with lOw-ranK AToms (DINO-KAT), and the dictionary model and images are simultaneously and sequentially estimated from streaming measurements. The proposed online algorithm involves efficient memory usage and simple and efficient updates of the low-rank atoms, sparse coefficients, and images. Our numerical experiments show the usefulness of the proposed scheme in inverse problem settings such as video reconstruction or inpainting from limited and noisy pixels.

***Index Terms*—** Sparse representations, dictionary learning, inverse problems, online algorithms, machine learning.

1. INTRODUCTION

Signal models involving sparsity, low-rank, and other properties have been widely used in image processing. Such models are especially important in inverse problem settings such as denoising, deblurring, inpainting, etc., where they are often used to construct regularizers that reflect known or assumed properties of data. While sparsity in wavelet or discrete cosine transform (DCT) domains has been extensively used in image and video restoration [1, 2], more recently, the data-driven adaptation of synthesis dictionary [3–5] or sparsifying transform [6–8] models has shown promise in numerous applications [9, 10]. Methods for online or sequential learning of such models from streaming measurements have also been proposed [4, 11, 12].

For dynamic data such as videos or dynamic magnetic resonance image (MRI) sequences, various models have been explored such as the well-known L+S model [13] that decomposes the dynamic image sequence into the sum of a low-rank (L) and a sparse (S) component. The L+S model has been shown to be useful for dynamic image reconstruction from compressive measurements [14]. There has also been growing interest in dictionary learning-based dynamic image reconstruction or restoration methods [15–17]. For example, in blind compressed sensing [10,16], the dictionary for the underlying image or video is assumed unknown, and it is estimated together with the image from undersampled measurements. Recently, the learning of a structured DIctioNary with lOw-ranK (upon reshaping the dictionary atoms into matrices) AToms (DINO-KAT) was explored [17]. Such structured dictionaries are particularly useful in applications involving limited or corrupted data (e.g., they are less prone to overfitting) and provide better reconstructions in scenarios such as blind compressed sensing compared to methods such as L+S [17].

For inverse problems involving large-scale or streaming data (e.g., interventional imaging or inpainting large/streaming videos), it is often critical to obtain reconstructions in an online or time-sequential manner to limit latency, and batch methods (that process all the data at once) are typically prohibitive or infeasible in time and memory. Hence, in this work, we investigate a framework for online data-driven reconstruction of dynamic image sequences from linear (typically undersampled) measurements. In particular, the spatiotemporal patches of the underlying dynamic image sequence are assumed to be sparse in a (unknown) DINO-KAT model, and the dictionary, sparse codes, and images are jointly and sequentially estimated from streaming measurements. The proposed online DINO-KAT-driven reconstruction algorithm involves simple and efficient updates and modest memory requirements. Our numerical experiments demonstrate the effectiveness of the online scheme and the suitability of adapting dictionaries with low-rank (reshaped) atoms for video inpainting from limited and noisy pixels.

2. PROBLEM FORMULATION AND ALGORITHM

This section presents our problem formulation and algorithm for online DINO-KAT-adaptive dynamic image reconstruction. In the following, we first briefly discuss the idea of learning DINO-KAT models from data.

2.1. Learning DINO-KAT Models

Given a set of signals (e.g., vectorized image patches) that are represented as columns of a matrix $P \in \mathbb{C}^{n \times M}$, the goal of dictionary learning (DL) is to learn a dictionary $D \in \mathbb{C}^{n \times m}$ and a matrix $Z \in \mathbb{C}^{m \times M}$ of sparse codes such that $P \approx DZ$. The DINO-KAT learning problem formulation is [17]

$$\text{(P0)}\ \min_{D,Z} \ \left\| P - DZ \right\|_F^2 + \lambda^2 \left\| Z \right\|_0$$
$$\text{s.t. } \operatorname{rank}\left(R(d_i)\right) \leq r,\ \left\| d_i \right\|_2 = 1,\ \left\| z_l \right\|_\infty \leq L \ \forall i, l,$$

where d_i and z_l denote the ith and lth columns of D and Z, respectively, and the ℓ_0 “norm" counts the number of non-zero entries of its argument (vector or matrix). The columns of D are set to unit norm to avoid the scaling ambiguity [18]. The operator $R(\cdot)$ reshapes dictionary atoms into matrices, and $r > 0$ is the maximum allowed rank of the reshaped atoms. For example, in the case of spatiotemporal (3D) patches of dynamic data, the atoms would be reshaped into space-time (2D) matrices. Spatiotemporal patches of videos typically have correlations along time, and thus they may be well-represented by a dictionary of low-rank space-time (reshaped) atoms.

This work was supported in part by the following grants: ONR grant N00014-15-1-2141, DARPA Young Faculty Award D14AP00086, ARO MURI grants W911NF-11-1-0391 and 2015-05174-05, NIH grants R01 EB 023618 and P01 CA 059827, and a UM-SJTU seed grant.

978-1-5090-2175-8/17/$31.00 ©2017 IEEE

The weight λ^2 with $\lambda > 0$ in (P0) controls the overall sparsity of the matrix Z, and enables variable sparsity levels across signals. The ℓ_∞ constraints (with $L > 0$) prevent pathologies that could theoretically arise (e.g., unbounded algorithm iterates) due to the objective being non-coercive [19]. In practice, we set L very large, and the constraint is inactive.

2.2. Online Adaptive Image Reconstruction Formulation

We propose an online image reconstruction framework based on an adaptive DINO-KAT regularizer (as in (P0)). Let $\left\{g^t \in \mathbb{C}^{N_x \times N_y}\right\}$ denote the sequence of dynamic image frames to be reconstructed. We assume that linear measurements of these frames are observed. Let x^t denote the vectorized version of the 3D array obtained by (temporally) stacking a small number $\tilde{M}$ of consecutive frames. We construct sequence $\left\{x^t\right\}$ using a sliding window (over time) strategy. We assume that the spatiotemporal (3D) patches in each x^t are sparse in some adaptive dictionary domain. We then solve the following DINO-KAT-driven (online) image reconstruction problem for each index $t = 1, 2, 3$, etc.

$$\text{(P1)}\ \left\{\hat{x}^t, \hat{D}^t, \hat{Z}^t\right\} = \underset{x^t, D, Z^t}{\arg\min}\ \frac{1}{2K_t} \sum_{j=1}^{t} \rho^{t-j} \left\| y^j - A^j x^j \right\|_2^2$$
$$+ \frac{\lambda_S}{K_t} \sum_{j=1}^{t} \rho^{t-j} \Big(\sum_{l=1}^{M} \left\| P_l x^j - D z_l^j \right\|_2^2 + \lambda_Z^2 \left\| Z^j \right\|_0 \Big)$$
$$\text{s.t.}\ \ \text{rank}\left(R(d_i)\right) \leq r,\ \left\|d_i\right\|_2 = 1,\ \left\|z_l^t\right\|_\infty \leq L,\ \forall\, i, l,$$
$$x^j = \hat{x}^j,\ Z^j = \hat{Z}^j,\ 1 \leq j \leq t-1.$$

Here, j indexes time, the operator A^t is the sensing/encoding operator corresponding to the frames in x^t, and y^t denotes the corresponding (typically undersampled) measurements. For example, in video inpainting, A^t samples a subset of pixels in x^t, and in dynamic MRI, it corresponds to a (undersampled) Fourier encoding. The operator P_l is a patch extraction matrix that extracts an $n_x \times n_y \times n_t$ spatiotemporal patch from x^t as a vector. A total of M overlapping 3D patches are assumed for x^t. Matrix $D \in \mathbb{C}^{n \times m}$ with $n = n_x n_y n_t$ is the synthesis dictionary to be learned, and $z_l^t \in \mathbb{C}^m$ is the unknown sparse code for the lth patch of x^t, with $P_l x^t \approx D z_l^t$. Matrix Z^t has z_l^t as its columns. The weights λ_S and λ_Z are non-negative.

Problem (P1) estimates the DINO-KAT model for image patches together with the underlying image frames. For each time index t, we only solve for the latest group of frames x^t and the latest sparse coefficients Z^t (i.e., online optimization). However, the dictionary D is adapted to all the spatiotemporal patches observed up to time t. A (exponential) forgetting factor ρ^{t-j} with $0 < \rho < 1$ is used to weight the terms in (P1), with $K_t = \sum_{j=1}^{t} \rho^{t-j}$ being a normalization constant for the overall cost. The forgetting factor diminishes the influence of "old" data on the dictionary adaptation process. When the dynamic object or scene changes slowly over time, a large ρ (close to 1) is preferable so that past information has a relatively larger impact on the cost. Although the dictionary D is updated based on all patches in (P1), our algorithm proposed in Section 2.3 does not require storing all the data over time; it only stores a few constant-sized (over time) matrices that contain the necessary cumulative information to minimize the cost.

When x^t and x^{t+1} do not overlap (i.e., no common frames between them), each frame g^t is reconstructed once (in its corresponding window of $\tilde{M}$ frames) via (P1). However, it is often beneficial to construct the x^t's using an overlapping sliding window strategy [12]. In this case, we independently reconstruct the frame g^t in each of the windows in which it occurs in (P1). These instantaneous window-wise reconstructions are then averaged together according to the forgetting factors in (P1) to obtain the final frame estimates. We empirically found that such an exponentially-ρ-weighted reconstruction performed better than alternatives such as an unweighted average (of estimates from multiple windows) or using the instantaneous estimate from the latest window.

2.3. Algorithm and Properties

We propose a simple iterative procedure for (P1) for each time index t. The algorithm is initialized with the most recently estimated dictionary, $\hat{D}^{t-1}$. Frames of x^t that were estimated in the previous (temporal) windows are initialized with the most recent (ρ-)weighted reconstruction, whereas new frames are initialized using simple approaches (e.g., interpolation in the case of inpainting). Initializing the sparse coefficients Z^t with the codes estimated in the preceding window worked well. The iterative procedure for (P1) alternates (only) a few times between updating $\left(D^t, Z^t\right)$ (*dictionary learning step*) and x^t (*image update step*). All updates are performed efficiently and with modest memory usage as detailed next.

2.3.1. Dictionary Learning Step

When minimizing (P1) with respect to (D, Z^t), we set $C^t = (Z^t)^H$ and arrive at the following optimization problem:

$$\text{(P2)}\ \min_{D, C^t}\ \sum_{j=1}^{t} \rho^{t-j} \left\| P^j - D (C^j)^H \right\|_F^2 + \lambda_Z^2 \left\| C^t \right\|_0$$
$$\text{s.t.}\ \ \text{rank}\left(R(d_i)\right) \leq r,\ \left\|d_i\right\|_2 = 1,\ \left\|c_i^t\right\|_\infty \leq L,\ \forall\, i,$$

where P^j has the patches $P_l x^j$ ($1 \leq l \leq M$) as its columns, and c_i^t is the ith column of C^t. We use a block coordinate descent approach to update the coefficient columns c_i^t (of C^t) and atoms d_i (of D) sequentially. In particular, for each $1 \leq i \leq m$, we first solve (P2) with respect to c_i^t keeping the other variables fixed (*sparse coding step*), and then update d_i (*dictionary atom update step*).

Sparse Coding Step. Here, we minimize (P2) with respect to c_i^t, which leads to the following problem, where the matrix $E_i^t \triangleq P^t - \sum_{k \neq i} d_k (c_k^t)^H$ involves the most recent estimates of the other atoms and coefficients:

$$\min_{c_i^t \in \mathbb{C}^M}\ \left\| E_i^t - d_i (c_i^t)^H \right\|_F^2 + \lambda_Z^2 \left\| c_i^t \right\|_0\ \ \text{s.t.}\ \ \left\| c_i^t \right\|_\infty \leq L. \quad (1)$$

The solution, assuming $L > \lambda_Z$, is given as [19]

$$\hat{c}_i^t = \min\left(\left| H_{\lambda_Z}\left((E_i^t)^H d_i \right) \right|, L 1_M \right) \odot e^{j \angle (E_i^t)^H d_i}, \quad (2)$$

where $H_{\lambda_Z}(\cdot)$ is the hard thresholding operator that sets vector entries with magnitude $< \lambda_Z$ to zero, 1_M is a vector of ones of length M, "$\odot$" denotes element-wise multiplication, $\min(\cdot, \cdot)$ denotes element-wise minimum, and $e^{j \angle q}$ for vector $q \in \mathbb{C}^M$ is computed element-wise, with "$\angle$" denoting the phase. We do not construct E_i^t explicitly, rather the vector $(E_i^t)^H d_i = (P^t)^H d_i - C^t D^H d_i + c_i^t$ based on the most recent estimates of variables, is computed using cheap operations (such as sparse matrix-vector operations) [19].

Dictionary Atom Update Step. Here, we minimize (P2) with respect to d_i. This update makes use of past information via the forgetting factor, ρ. Let $\tilde{P}^j \triangleq \sqrt{\rho^{t-j}} P^j$, $\tilde{C}^j \triangleq \sqrt{\rho^{t-j}} C^j$, and $\tilde{P}^{1:t}$ and $\tilde{C}^{1:t}$ denote matrices obtained by stacking the $\tilde{P}^j$'s (horizontally) and $\tilde{C}^j$'s (vertically) respectively for time 1 to t. Furthermore, let $\tilde{E}_i^{1:t} \triangleq \tilde{P}^{1:t} - \sum_{k \neq i} d_k (\tilde{c}_k^{1:t})^H$ be a matrix obtained using the most recent estimates of all variables, with $\tilde{c}_k^{1:t}$ denoting the kth column of $\tilde{C}^{1:t}$. Then, we have the following atom update sub-problem:

$$\min_{d_i \in \mathbb{C}^n} \left\| \tilde{E}_i^{1:t} - d_i (\tilde{c}_i^{1:t})^H \right\|_F^2 \;\text{ s.t. } \operatorname{rank}\left(R(d_i)\right) \leq r, \; \|d_i\|_2 = 1. \tag{3}$$

Let $U_r \Sigma_r V_r^H$ be the rank-r approximation to $R(\tilde{E}_i^{1:t} \tilde{c}_i^{1:t})$ that is obtained using the r leading singular vectors and singular values of the singular value decomposition (SVD) $R(\tilde{E}_i^{1:t} \tilde{c}_i^{1:t}) \triangleq U \Sigma V^H$. Then a global minimizer of (3), upon reshaping, is [17]

$$R(\hat{d}_i) = \begin{cases} \frac{U_r \Sigma_r V_r^H}{\|\Sigma_r\|_F}, & \text{if } \tilde{c}_i^{1:t} \neq 0 \\ W_0, & \text{if } \tilde{c}_i^{1:t} = 0, \end{cases} \tag{4}$$

where W_0 is any normalized rank $\leq r$ matrix of appropriate size (e.g., we use the reshaped first column of the $n \times n$ identity matrix).

The main computation in (4) is computing $\tilde{E}_i^{1:t} \tilde{c}_i^{1:t}$, while computing the SVD of the (small) matrix has negligible computational cost. In particular, $\tilde{E}_i^{1:t} \tilde{c}_i^{1:t} = \tilde{E}_i^{1:t-1} \tilde{c}_i^{1:t-1} + E_i^t c_i^t$, where c_i^t is as estimated in (2). First, $E_i^t c_i^t$ can be computed cheaply using sparse matrix-vector operations, as $E_i^t c_i^t = P^t c_i^t - D(C^t)^H c_i^t + d_i (c_i^t)^H c_i^t$, where C^t includes the estimate (2). However, $\tilde{E}_i^{1:t-1} \tilde{c}_i^{1:t-1} = \tilde{P}^{1:t-1} \tilde{c}_i^{1:t-1} - D(\tilde{C}^{1:t-1})^H \tilde{c}_i^{1:t-1} + d_i \left\| \tilde{c}_i^{1:t-1} \right\|_2^2$ relies on past information. Rather than storing this information, we recursively pre-compute the following *small* matrices at each time t using sparse matrix operations:

$$\begin{aligned} \tilde{P}^{1:t-1} \tilde{C}^{1:t-1} &\leftarrow \rho P^{t-1} C^{t-1} + \rho (\tilde{P}^{1:t-2} \tilde{C}^{1:t-2}) \\ (\tilde{C}^{1:t-1})^H \tilde{C}^{1:t-1} &\leftarrow \rho (C^{t-1})^H C^{t-1} + \rho \left((\tilde{C}^{1:t-2})^H \tilde{C}^{1:t-2} \right), \end{aligned}$$

where the second terms on the right hand side above correspond to the previous (from time $t - 1$) estimates of these matrices. Using these quantities, $\tilde{E}_i^{1:t-1} \tilde{c}_i^{1:t-1}$ can be readily computed in our algorithm. Thus, the update in (4) can be performed fully online.

2.3.2. *Image Update Step*

Minimizing (P1) with respect to x^t yields the sub-problem:

$$\text{(P3)} \quad \min_{x^t} \frac{1}{2} \left\| y^t - A^t x^t \right\|_2^2 + \lambda_S \sum_{l=1}^{M} \left\| P_l x^t - D z_l^t \right\|_2^2 .$$

The x^t minimizing the above (least squares) problem satisfies the following normal equation, where $(\cdot)^H$ denotes matrix Hermitian:

$$\left((A^t)^H A^t + 2\lambda_S \sum_{l=1}^{M} P_l^H P_l \right) x^t = (A^t)^H y^t + 2\lambda_S b^t, \tag{5}$$

where $b^t = \sum_{l=1}^{M} P_l^H D z_l^t$. In applications such as video denoising or inpainting where $(A^t)^H A^t$ is diagonal, the matrix pre-multiplying x^t in (5) is diagonal and can be pre-computed. The solution is then obtained using simple patch-based operations for b^t. In inverse problems, where the matrix pre-multiplying x^t in (5) isn't diagonal or readily diagonalizable (e.g., multi-coil dynamic MRI), we instead use a few iterations (indexed by k) of the proximal gradient scheme for (P3), which take the form

$$x^{t,k} = \operatorname{prox}_{t_k h} \left(x^{t,k-1} - t_k (A^t)^H (A^t x^{t,k-1} - y^t) \right), \tag{6}$$

where $h(x^t) \triangleq \lambda_S \sum_{l=1}^{M} \left\| P_l x^t - D z_l^t \right\|_2^2$ and the proximity function is defined as $\operatorname{prox}_{t_k h}(x) = \arg\min_{z} \; 0.5 \|x - z\|_2^2 + t_k \, h(z)$. The proximity mapping for (6) involves solving a normal equation by inverting a diagonal (pre-computable) matrix, and can be performed cheaply. A constant step-size $t_k = t < 2 / \left\| A^t \right\|_2^2$ suffices for the proximal gradient method to converge [20].

2.3.3. *Algorithm Properties*

The computational cost for each time index t of the proposed online image reconstruction algorithm scales as $O(n^2 M)$ for $D \in \mathbb{C}^{n \times m}$, $m \propto n$, and for M overlapping patches in each temporal window (each x^t). The cost is dominated by various matrix-vector multiplications. Assuming the window length $\tilde{M} \ll n$, the proposed algorithm has a memory (storage) cost scaling as $O(nM)$ for storing the patches (while performing updates for (P1)) in each temporal window. The number of 3D patches is typically small for practical window lengths $\tilde{M}$ and typical image (frame) sizes, which ensures modest memory usage for our online method.

3. NUMERICAL EXPERIMENTS

Here, we illustrate the usefulness of the proposed online DINO-KAT learning-driven dynamic image reconstruction method for video reconstruction (inpainting) from noisy and subsampled pixels. We work with the publicly available gray-scale road video.[1] We resize the 480 × 640 × 396 video to size 128 × 171 × 200 (keeping the first 200 frames and resizing each frame) in our experiments. The pixel intensities are scaled to the range [0, 1]. We measure a random subset of pixels in each frame of the video, and also simulate (Gaussian) noise in the measured pixels. The proposed online data-driven image reconstruction approach is used to reconstruct the video from its corrupted (noisy and undersampled) measurements.

For the proposed method, we use temporal (sliding) windows of length $\tilde{M} = 5$ and a temporal stride of 1 frame. In each window, $8 \times 8 \times 5$ overlapping spatiotemporal patches are extracted with a spatial stride of 1 pixel. We learn a square 320×320 dictionary, and the operator $R(\cdot)$ reshapes dictionary atoms into 64×5 (space-time) matrices. The algorithm for (P1) ran for 10 iterations (alternations) in each temporal window, with 1 inner iteration of block coordinate descent for updating (D, Z^t), and $\rho = 0.8$. The algorithm ran for more (30) iterations for the first 5 frames (first window), for which D was initialized with a DCT dictionary, the sparse codes with zero, and the frames with 2D (cubic) interpolated results. We simulated various levels of subsampling of the video (with and without noise), and the proposed algorithm was simulated with atom ranks $r = 1, 3, 5$. The weights λ_S and λ_Z were tuned at an intermediate undersampling factor (70% missing pixels) for the video.

We compare the performance of the proposed online adaptive reconstruction method at various atom ranks with that of 2D (per-frame cubic) interpolation, 3D interpolation (using the natural neighbor method in MATLAB), and the proposed online method with a *fixed* DCT dictionary. Table 1 lists the peak signal-to-noise ratio (PSNR) values in decibels (dB) for the various reconstruction methods at different levels of subsampling (from 50% to 90% missing pixels) of the road video. Table 2 shows results when the sampled pixels are also noisy. In the latter case, the fully-sampled video had a PSNR of 25 dB. The proposed online DINO-KAT learning-driven image reconstruction method provides the best PSNRs in the tables. 3D interpolation outperforms 2D interpolation, and the DINO-KAT approach outperforms these as well as the method employing the fixed DCT dictionary. Importantly, the proposed method with low-rank atoms ($r = 1, 3$) outperforms the full-rank ($r = 5$) case. The

[1] This is the Sequence 2 video at `http://ccv.wordpress.fos.auckland.ac.nz/eisats/set-2/`.

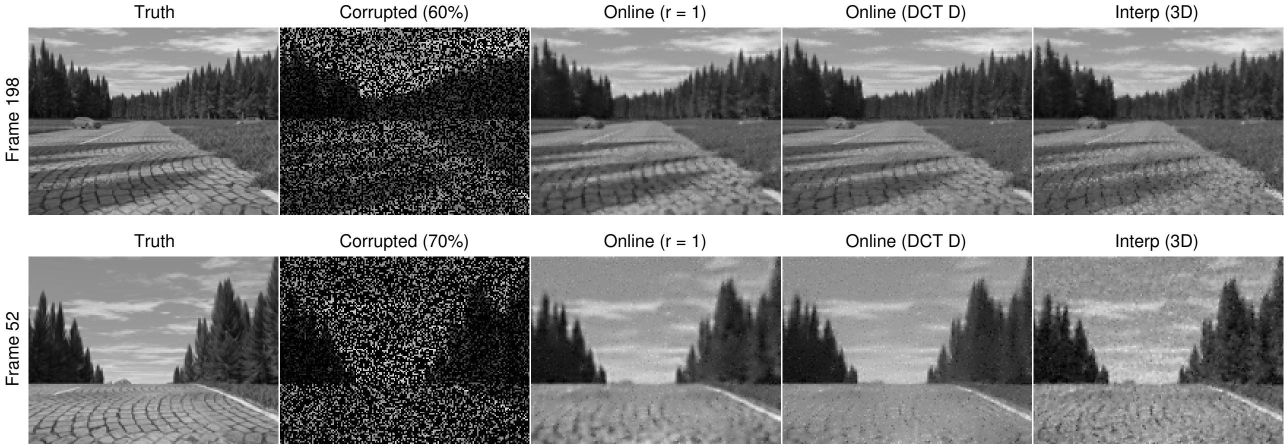

Fig. 1. Results for the proposed online DINO-KAT learning-driven reconstruction method ($r = 1$), the online method with fixed DCT D, and 3D interpolation. *Top*: Reconstructions of frame 198 from 60% missing pixels and no noise: DINO-KAT provides 0.03 dB and 2.25 dB better PSNR than the online DCT-based and 3D interpolation methods. *Bottom*: Reconstructions of frame 52 from noisy measurements and 70% missing pixels: DINO-KAT achieves 0.64 dB and 1.25 dB better PSNR than the online DCT-based and 3D interpolation methods.

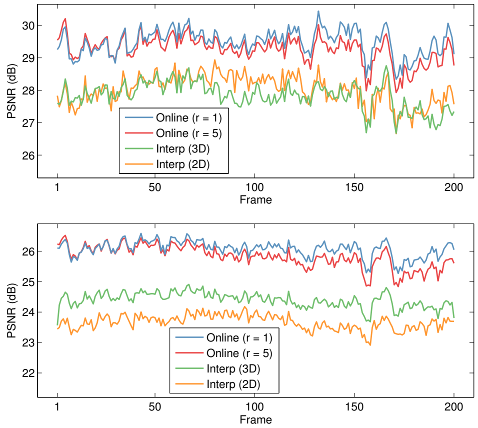

Fig. 2. PSNR values between each reconstructed and reference video frame for the case of 50% missing pixels, with 2D (per-frame cubic) interpolation, 3D interpolation, and with the proposed online DINO-KAT learning-driven inpainting with atom ranks $r = 1$ and $r = 5$. The plots are for the noiseless (top) and noisy (bottom) pixel cases. The fully-sampled noisy video has a PSNR of 25 dB.

average (in Tables 1 and 2) PSNR improvements provided by the DINO-KAT method with $r = 1$ over 2D interpolation, 3D interpolation, the online non-adaptive (DCT) scheme, and the online data-driven scheme with full-rank ($r = 5$) reshaped atoms are 2.05 dB, 0.99 dB, 0.58 dB, and 0.15 dB, respectively.

Figure 1 shows some original and reconstructed frames for the online DINO-KAT based method ($r = 1$), online DCT-based scheme, and 3D interpolation. The reconstructed frames for the proposed scheme are much closer to the reference than for the others. Figure 2 shows the per-frame PSNRs for the inpainted videos using various methods, clearly demonstrating the benefit provided by the online DINO-KAT scheme with $r = 1$. We observed that the adaptive dictionary estimates converged (over time) as more frames were processed, and correspondingly, the PSNR gap between $r = 1$ and $r = 5$ grows bigger after some initial frames in Figure 2. More extensive inpainting results and comparisons will be presented in future work.

% Missing Pixels	50%	60%	70%	80%	90%
Interpolation (2D)	28.28	26.83	25.33	23.66	21.45
Interpolation (3D)	28.10	26.98	25.92	24.80	23.44
Online (DCT D)	29.56	28.01	26.46	24.78	22.71
Online DINO-KAT ($r = 5$)	29.52	28.06	26.67	25.23	23.57
Online DINO-KAT ($r = 3$)	29.63	28.16	26.78	25.28	**23.58**
Online DINO-KAT ($r = 1$)	**29.74**	**28.25**	**26.88**	**25.36**	23.52

Table 1. PSNR values in dB for video inpainting at various percentages of missing pixels for 2D (per-frame cubic) interpolation, 3D interpolation, online inpainting with a fixed DCT dictionary, and the proposed online DINO-KAT learning-driven method with rank $r = 1, 3, 5$ atoms. The best PSNR at each undersampling is in bold.

% Missing Pixels	50%	60%	70%	80%	90%
Interpolation (2D)	23.95	23.46	22.80	21.86	20.27
Interpolation (3D)	24.68	24.42	24.05	23.50	22.60
Online (DCT D)	26.16	25.42	24.49	23.30	21.71
Online DINO-KAT ($r = 5$)	26.09	25.60	24.97	24.19	22.95
Online DINO-KAT ($r = 3$)	26.22	25.67	25.00	24.21	22.96
Online DINO-KAT ($r = 1$)	**26.35**	**25.79**	**25.16**	**24.30**	**23.03**

Table 2. PSNRs in dB for video inpainting at various percentages of missing pixels and with noise in the measured pixels (25 dB PSNR). Results shown for 2D (per-frame cubic) interpolation, 3D interpolation, online inpainting with a fixed DCT dictionary, and the proposed online DINO-KAT learning-driven method with rank $r = 1, 3, 5$ atoms. The best PSNRs are in bold.

4. CONCLUSIONS

This paper presented a novel framework for online estimation of dynamic image sequences by learning dictionaries with low-rank atoms. The proposed algorithm sequentially and efficiently updates the dictionary, sparse coefficients of patches, and images from streaming measurements. We showed the promise of online DINO-KAT learning-driven image reconstruction for video inpainting from limited and corrupted pixels. Importantly, the adapted dictionaries with low-rank atoms provided improved reconstruction quality compared to dictionaries with full-rank atoms. The proposed approach could be promising for other applications (e.g., interventional medical imaging), which we hope to explore in future work.

5. REFERENCES

[1] N. Rajpoot, Z. Yao, and R. Wilson, "Adaptive wavelet restoration of noisy video sequences," in *International Conference on Image Processing*, 2004, vol. 2, pp. 957–960 Vol.2.

[2] D. Rusanovskyy and K. Egiazarian, "Video denoising algorithm in sliding 3D DCT domain," in *Advanced Concepts for Intelligent Vision Systems: 7th International Conference*, 2005, pp. 618–625.

[3] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[4] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.

[5] M. Yaghoobi, T. Blumensath, and M. Davies, "Dictionary learning for sparse approximations with the majorization method," *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2178–2191, 2009.

[6] S. Ravishankar and Y. Bresler, "Learning sparsifying transforms," *IEEE Transactions on Signal Processing*, vol. 61, no. 5, pp. 1072–1086, 2013.

[7] S. Ravishankar and Y. Bresler, "Learning doubly sparse transforms for images," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4598–4612, 2013.

[8] B. Wen, S. Ravishankar, and Y. Bresler, "Learning flipping and rotation invariant sparsifying transforms," in *IEEE International Conference on Image Processing*, 2016, pp. 3857–3861.

[9] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.

[10] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1028–1041, 2011.

[11] S. Ravishankar, B. Wen, and Y. Bresler, "Online sparsifying transform learning – Part I: Algorithms," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 625–636, 2015.

[12] B. Wen, S. Ravishankar, and Y. Bresler, "Video denoising by online 3D sparsifying transform learning," in *IEEE International Conference on Image Processing*, 2015, pp. 118–122.

[13] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *Journal of the ACM*, vol. 58, no. 3, pp. 11:1–11:37, 2011.

[14] R. Otazo, E. Candès, and D. K. Sodickson, "Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components," *Magnetic Resonance in Medicine*, vol. 73, no. 3, pp. 1125–1136, 2015.

[15] M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 27–36, 2009.

[16] S. G. Lingala and M. Jacob, "Blind compressive sensing dynamic MRI," *IEEE Transactions on Medical Imaging*, vol. 32, no. 6, pp. 1132–1145, 2013.

[17] S. Ravishankar, B. E. Moore, R. R. Nadakuditi, and J. A. Fessler, "Efficient learning of dictionaries with low-rank atoms," in *IEEE Global Conference on Signal and Information Processing*, 2016, pp. 222–226.

[18] R. Gribonval and K. Schnass, "Dictionary identification–sparse matrix-factorization via l_1 -minimization," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3523–3539, 2010.

[19] S. Ravishankar, R. R. Nadakuditi, and J. A. Fessler, "Efficient sum of outer products dictionary learning (SOUP-DIL) and its application to inverse problems," *IEEE Transactions on Computational Imaging*, 2017, to appear.

[20] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.