



Non-parametric Ensemble Kalman methods for the inpainting of noisy dynamic textures

Redouane Lguensat, Pierre Tando, Ronan Fablet, Pierre Ailliot

► To cite this version:

Redouane Lguensat, Pierre Tando, Ronan Fablet, Pierre Ailliot. Non-parametric Ensemble Kalman methods for the inpainting of noisy dynamic textures. ICIP 2015: IEEE International Conference on Image Processing, Sep 2015, Québec City, Canada. pp.2488-2492, 10.1109/ICIP.2015.7351615 . hal-01271173

HAL Id: hal-01271173

<https://hal.science/hal-01271173>

Submitted on 8 Feb 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

NON-PARAMETRIC ENSEMBLE KALMAN METHODS FOR THE INPAINTING OF NOISY DYNAMIC TEXTURES

Redouane Lguensat, Pierre Tandeo, Ronan Fablet

Institut Mines-Telecom; Telecom Bretagne
CNRS UMR 6285 Lab-STICC
Brest, France.

Pierre Ailliot

University of Western Brittany
CNRS UMR 6205 LMBA
Brest, France.

ABSTRACT

In this work, we propose a novel non parametric method for the temporally-consistent inpainting of dynamic texture sequences. The inpainting of texture image sequences is stated as a stochastic assimilation issue, for which a novel model-free and data-driven Ensemble Kalman method is introduced. Our model is inspired by the Analog Ensemble Kalman Filter (AnEnKF) recently proposed for the assimilation of geophysical space-time dynamics, where the physical model is replaced by the use of statistical analogs or nearest neighbours. Such a non-parametric framework is of key interest for image processing applications, as prior models are seldom available in general. We present experimental evidence for real dynamic texture that using only a catalog database of historical data and without having any assumption on the model, the proposed method provides relevant dynamically-consistent interpolation and outperforms the classical parametric (autoregressive) dynamical prior.

Index Terms— Data assimilation, Dynamic textures, Inpainting, Ensemble kalman filter, Nearest-Neighbors, Data mining.

1. INTRODUCTION

We are interested in the analysis of dynamic textures (DTs), which can be defined as image sequences characterized by repetitive space-time patterns. Examples of dynamic textures include sea waves, fire flames, moving flags, etc... The interested reader can refer to [1] and the references therein for characterizations and examples of DTs. Recent studies have mainly focused on DT modelling, synthesis and recognition. Here, we address the dynamically-consistent reconstruction of DT sequence from noisy observations, possibly involving missing data.

Let $x(t_k)_{k=1\dots\tau}$ be the sequence of images to be inferred and suppose we have $y(t_k)_{k=1\dots\tau}$ incomplete and noisy measurements of $x(t_k)$ at each time k . This inverse problem can be formulated by the following equation:

$$y(t_k) = Hx(t_k) + \epsilon(t_k) \quad (1)$$

where H is an observation operator representing a mask (0 for unavailable data, 1 for presence) and ϵ a random perturbation which takes into consideration observation modeling error, acquisition noise and other sources of uncertainty. These measurements are one source of information about the sequence $x(t)$.

Within a classical Bayesian state-space setting (REF), a second source of information about this sequence is provided by the dynamical model that controls the evolution of the DT in time:

$$x(t_{k+1}) = \mathcal{M}(x(t_k)) + \eta(t_k) \quad (2)$$

where \mathcal{M} denotes the dynamical model and η a random noise representing the uncertainty about the model between two consecutive times. The mathematical resolution of the state-space model (1)-(2) refers to data assimilation in the geoscience community or more commonly stochastic filtering.

Here, we focus on a flexible and efficient sequential Monte Carlo technique: the Ensemble Kalman method (see [2] for a review of widely used data assimilation methods). In its classical form, it proceeds sequentially as follows. One first runs the dynamical model \mathcal{M} given in Eq.(2) at each time step to simulate ensemble members corresponding to representative set of forecast scenarios. The second step amounts to reweighting the different members according to their agreement to the new observation. This classical setting requires the knowledge of the dynamical model, what may be complex especially for visual image sequences for which no underlying fluid-dynamics-like interpretation can be derived. By contrast, we exploit a new data-driven methodology which makes use of a database of model-simulated or observation-based scenarios to emulate the dynamical model: this recent methodology is called the Analog Ensemble Kalman Filter and Smoother (AnEnKF and AnEnKS) and has been introduced in [4]. In this previous work, experiments using Analog ensemble methods were done in the Lorenz-63 model in which the state of the system is a 3-dimensional vector [5]. The results indicate that the AnEnKF and AnEnKS are able to retrieve the chaotic behaviour of such model. Our contri-

bution consists in applying and extending this non-parametric method to reconstruct a priori unknown complex dynamic textures from partial and noisy observations sequence and addresses the curse of dimensionality.

The paper is organized as follows. Section II presents the AnEnKF/AnEnKS algorithm. Section III describes our DTs data and the dimensionality reduction method. In Section IV, the application to a reference DTs database is evaluated. We further discuss and summarize the key results of our investigations in Section V.

2. ANALOG ENSEMBLE KALMAN FILTER/SMOOTHER (ANENKF/ANENKS)

Limitations of stochastic data assimilation and especially difficulties faced to explicitly model the dynamics and state the parametrizations of the system were the motivation behind developing the AnEnKF/AnEnKS [5]. The core idea consists in combining machine learning and stochastic filtering methods. It exploits previously acquired datasets which contain examples of the time evolution of the state and explores implicit data-driven models. In the subsequent we assume a *catalog* is built from previously observed states of the system (referred to as analogs) along with the associated following states (successors).

2.1. Nonparametric sampler of the dynamics

We consider a non-parametric sampling of state dynamics using a nearest-neighbour scheme. Let us consider $x(t)$ to be the state of the system at time t . To generate possible forecast states at time $t + dt$, we search for the nearest neighbors (analogs) of the state in the catalog. This is a classic method used in data mining [9] that will help us to get to their matching successors. Then, we use the weighted K-nearest neighbors (WKNN [10]) to compute a realistic prediction of the future state (successor). This non-parametric data-driven sampling of the state dynamics is plugged into a classical ensemble data assimilation method. In this work, we use the Ensemble Kalman Filter (EnKF) [3]. Algorithm 1 illustrates the proposed non-parametric sampler for the statistical emulation of the dynamical systems \mathcal{M} .

2.2. Ensemble Kalman Filter and Smoother

In the initial step of the EnKF algorithm, at time $t = 1$, we construct the vectors $\mathbf{x}_i^f(1) \forall i \in \{1, \dots, N\}$ using a multivariate Gaussian random generator with the mean vector \mathbf{x}^b and the covariance matrix \mathbf{B} , these are *a priori* information called the background. Then, we proceed forward from $t = 2$ to $t = T$. In the update step, we apply the nonparametric sampler presented above to generate N samples of $\mathbf{x}_i^f(t)$ from a multivariate Gaussian random generator with weighted mean

Algorithm 1 Nonparametric sampler of the dynamics

- 1: **for** All previous analysis results $\mathbf{x}_i^a(t - dt)$, i.e. each member/particle $i \in \{1, \dots, N\}$ and for each time analysis $t \in \{1, \dots, T\}$ **do**
 - 2: Find the K nearest neighbors/analogues using a Kd-tree procedure
 - 3: Compute the corresponding K Euclidean distances with $\mathbf{x}_i^a(t - dt)$ and determine the corresponding k normalized weights using an exponential kernel
 - 4: Extract the K corresponding successors for a delayed time dt
 - 5: Use the K successors and the k normalized weights to compute the weighted mean $\hat{\mathbf{x}}_i^f(t)$ and covariance $\hat{\mathbf{P}}_i^f(t)$
 - 6: **end for**
-

$\hat{\mathbf{x}}_i^f(t)$ and covariance $\hat{\mathbf{P}}_i^f(t)$. The forecast state is represented by the sample mean $\mathbf{x}^f(t)$ and the sample covariance $\mathbf{P}^f(t)$. In the analysis step, we generate the N samples of $\mathbf{y}_i^f(t)$ from a multivariate Gaussian random generator with mean $\mathbf{H}\mathbf{x}_i^f(t)$ and covariance \mathbf{R} which is the covariance of error ϵ introduced in 1. Then, the N members of the ensemble are updated by the observations as $\mathbf{x}_i^a(t) = \mathbf{x}_i^f(t) + \mathbf{K}^a(t)(\mathbf{y}(t) - \mathbf{y}_i^f(t))$ where $\mathbf{K}^a(t) = \mathbf{P}^f(t)\mathbf{H}'(\mathbf{H}\mathbf{P}^f(t)\mathbf{H}' + \mathbf{R})^{-1}$ is the Kalman filter gain. Finally, the updated Gaussian analyzed state is represented by the sample mean $\mathbf{x}^a(t)$ and the sample covariance $\mathbf{P}^a(t)$. An illustration of a time step of the filtering scheme is given in Fig. 1.

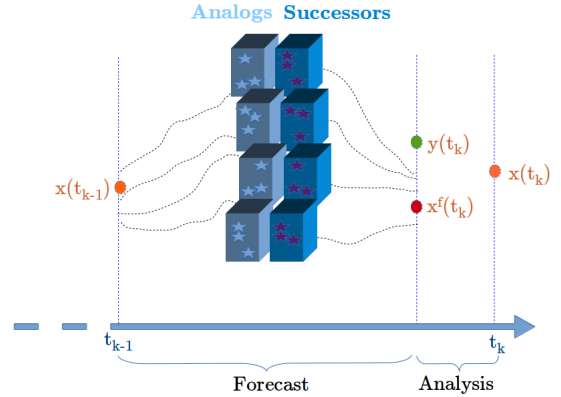


Fig. 1. Illustration of AnEnKF running in a time step: For each ensemble member, K nearest neighbors are retrieved from the analog and a weighted K -nearest neighbors is run on their corresponding successors, the mean of the results obtained for every ensemble member is the forecast state, the observation is then used in the analysis step with respect to EnKF equations.

The backward recursions correspond to the EnKS algorithm proposed by [12]. It uses the results of the EnKF com-

puted above. In the initial step of the EnKS algorithm, at time $t = T$, we use the members of the filtered state, $\forall i \in \{1, \dots, N\}$, such as $\mathbf{x}_i^s(T) = \mathbf{x}_i^a(T)$ and $\mathbf{P}^s(T) = \mathbf{P}^a(T)$. Then, we proceed backward from $t = T - 1$ to $t = 1$. At each time t , we compute $\mathbf{x}_i^s(t) = \mathbf{x}_i^a(t) + \mathbf{K}^s(t)(\mathbf{x}_i^s(t+1) - \mathbf{x}_i^f(t+1))$ where $\mathbf{K}^s(t) = \mathbf{P}^a(t)\mathcal{M}'(\mathbf{P}^f(t+1))^{-1}$ is the Kalman smoother gain. Finally, the updated Gaussian smoothed state is represented by the sample mean $\mathbf{x}^s(t)$ and the sample covariance $\mathbf{P}^s(t)$.

3. DATA DESCRIPTION AND DIMENSIONALITY REDUCTION

The DT sequences used in this study are from Soatto *et al.* paper [1], the database comprises various examples of DTs (escalator moving, human face talking, sea waves... etc), some of them were borrowed from the MIT temporal texture database [6] like the toilet flush sequence. Most of the examples are composed of 120 frames of size $n \times m$ where $100 \leq n, m \leq 200$, we use the first 100 frames as a training database and the 20 last as test images. Due to computational limits we can not use the image as a state of the model, so we proceed by using Principal Component Analysis (PCA) commonly used in computer vision (particularly in face recognition [7] [8]). Every image of the database is considered as a very long 1D vector by concatenating image pixels column by column, and we seek to derive eigenvectors of the covariance matrix of database image vectors, we borrow the term eigenfaces from face recognition to call them, next we choose $Nbfaces$ eigenfaces with the largest corresponding eigenvalues and we project the data on the lower-dimensional space formed by these eigenfaces.

The catalog of analogs and successors that we need for AnEnKF/AnEnKS is constructed by gathering the 1D columns of projections of every database image in an analogs' matrix, and putting the corresponding successors (which are for every image the next image in the video sequence) in a successors' matrix of the same dimension. Fig. 2 illustrates the scheme followed to construct the catalog.

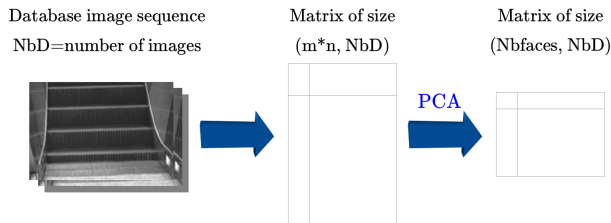


Fig. 2. Construction of the analogs' matrix, the successors' matrix is obtained by rearranging columns so that every analog has its successor at the same index, the example taken is the *escalator* sequence.

Thus to avoid computational costs, we consider the state vector used in AnEnKS to be the projection of the inferred image into the lower dimensional coordinate system (size of $x(t_k) = Nbfaces$). For the observations we faced two choices:

- Reducing observation dimension. The algorithm will be relatively simple and fast, but given that these are noncomplete and noisy, running PCA by eliminating missing data will cause a loss in the texture details (For further reading about methods of running PCA on data with missing values see [11]).
- Using the observation as it is and assimilating it into the model. This will result in heavy computational costs but will give intuitively better results than the previous choice because we do not compress the observations.

In this work we preferred simplicity and execution time and followed the first way, aiming at a proof concept of the methodology.

4. EXPERIMENTS

A Matlab[®] simulation was developed to test the algorithm on some noisy incomplete sequence of DTs. For every test image we added white noise then a square of random size and random position occulted a part of the image. The catalog is built by compressing the database images as explained in section III. For the considered sequences, about 100 Principal components were necessary to account for 99% of the total variance.

We run the AnEnKF/AnEnKS using 100 ensemble members for the EnKF and $k = 10$ nearest neighbors for the exhaustive search in the analog database. Our algorithm will be compared with reference assimilation model using a parametric autoregressive process. The AR(1) linearly models the temporal relationship between the analogs and the successors presented in the catalogs.

For each experiment, we compute the Root Mean Square Error (RMSE) between the true images (complete and without noise) and estimated smoothed states of DT after PCA reconstruction. RMSE will be calculated only on the indices of missing pixels in the observations. We vary the percentage of energy accounted for by PCA and analyze the impact on the data assimilation performance using two examples from the database: *toilet* and *escalator* sequences. Fig. 3 shows that our algorithm results in better RMSE when data are less compressed, the two sequences have redundancy in the database and thus results give a real-closed dynamic texture. Whereas Fig. 4 illustrates the difference map between the real image and results of the two algorithms. We see how our algorithm outperforms the AR model based one.

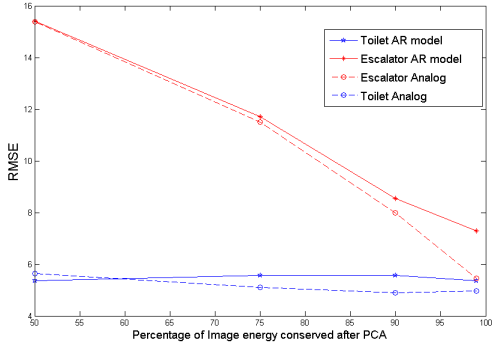


Fig. 3. Comparison between RMSE resulting from our data-driven based model algorithm and the AR model based one after a Monte Carlo simulation.

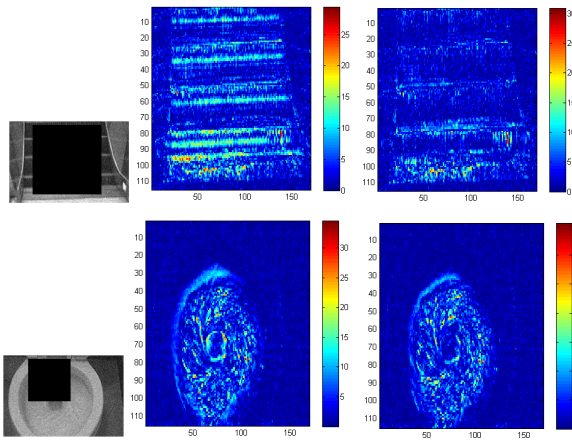


Fig. 4. From the left to the right: noisy and noncomplete observation. Difference map between the real image and result of AR model based algorithm. Difference map between the real image and result of our data-driven model based algorithm

Fig.5 shows visual results of the two algorithms on *face* sequence. We notice that for AR model based algorithm some inferred images are not closed to the true ones, and may be blurred or involve local artefacts. By contrast, our algorithm reconstructs relevant and realistic face images. The two algorithms also fail in some moments to retrieve the real inferred image. This is explained by the fact that the test images present a new scenario which does not exist in the database.

The smoothing effect presenting in the inferred images comes from the PCA compression and also from the WKNN algorithm, in future work we plan to consider the weights obtained after WKNN procedure and instead of multiplying them with the compressed versions of the database images, we will use directly the high resolution versions of database images.

5. CONCLUSION

This study presents a new methodology combining stochastic filtering and machine learning to handle denoising and image interpolation in the context of dynamic textures. We successfully retrieve complex dynamics of a sequence without any explicit parametric model but just a database containing previous scenarios. We have shown that our algorithm is better and gives more realistic results than autoregressive model based data assimilation algorithm. It demonstrates the feasibility of the non-parametric data-driven reconstruction of image dynamics, when the spatial image patterns can be projected onto a lower-dimensional space (up to ~ 100 dimensions).

Among others, future work will also investigate patch-based extensions of the proposed methodology. Patch-based models can be expected to improve applications to more complex spatial patterns, which may be complex to be accounted for using PCA-based decomposition. We will also investigate the use of other filtering methods especially particle filters and since the algorithm is appealing for fields where huge amount of data is available we aim at testing our model on more bigger database sizes such as those of remote sensing applications.



Fig. 5. *face* sequence. Algorithms are run conserving 99% of image energy for PCA. The first images are observations, second row shows results of AR model based data assimilation algorithm and the third row highlights results of our data-driven model based algorithm.

6. ACKNOWLEDGMENT

The authors would like to thank Stefano Soatto and Gianfranco Doretto for providing the DTs database.

7. REFERENCES

- [1] Doretto, G., Chiuso, A., Wu, Y. N., and Soatto, S. Dynamic textures International Journal of Computer Vision, 51(2):91109, 2003.
- [2] G. Evensen, Data assimilation. Springer, 2007.
- [3] Evensen, G. (2003). The ensemble Kalman filter: Theoretical formulation and practical implementation. Ocean dynamics, 53(4), 343-367.
- [4] P. Tandeo, P. Ailliot, R. Fablet, J. Ruiz, F. Rousseau, B. Chapron The Analog Ensemble Kalman Filter and Smoother. CI 2014 : 4th International Workshop on Climate Informatics, 25-26 september 2014, Boulder, United States, 2014.
- [5] P. Tandeo, P. Ailliot, J. Ruiz, A. Hannart, B. Chapron, A. Cuzol, V. Monbet, R. Easton and R. Fablet, Combining analog method and ensemble data assimilation: application to the Lorenz-63 chaotic system. Machine Learning and Data Mining Approaches to Climate Science, to appear.
- [6] M. Szummer, R. W. Picard. Temporal Texture Modeling September 1996 Lausanne, Switzerland IEEE Intl Conf On Image Processing (ICIP) pp. 823-826 vol. 3
- [7] Turk, M., Pentland, A. (1991). Eigenfaces for recognition. Journal of cognitive neuroscience, 3(1), 71-86.
- [8] Belhumeur, P. N., Hespanha, J. P., Kriegman, D. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 19(7), 711-720.
- [9] Friedman JH, Bentley JL, Finkel RA. (1977) An algorithm for finding best matches in logarithmic expected time. ACM Transactions on Mathematical Software (TOMS) 3(3):209-226.
- [10] Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. Systems, Man and Cybernetics, IEEE Transactions on, (4), 325-327.
- [11] Dray, S., Josse, J. (2014). Principal component analysis with missing values: a comparative survey of methods. Plant Ecology, 1-11.
- [12] Evensen, G., Van Leeuwen, P. J. (2000). An ensemble Kalman smoother for nonlinear dynamics. Monthly Weather Review, 128(6), 1852-1867.