

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

7,200

Open access books available

192,000

International authors and editors

210M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

14%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter

Principal Components and Factor Models for Space-Time Data of Remote Sensing

Carlo Grillenzoni

Abstract

Time-lapse videos, created with sequences of remotely-sensed images, are widely available nowadays; their aim is monitoring land transformations, both as regards natural events (e.g., floods) and human interventions (e.g., urbanizations). The corresponding datasets are represented by multidimensional arrays (at least 3-4D) and their spectral analysis (eigenvalues, eigenvectors, principal components, factor models) poses several issues. In particular, one may wonder what are the statistically meaningful operations and what is the treatment of the space-time autocorrelation (ACR) across pixels. In this article, we develop principal component analysis (PCA, useful for data reduction and description) and factor autoregressive models (FAR, suitable for data analysis and forecasting), for 3D data arrays. An extensive application, to a real case study of a Google Earth video, is carried out to illustrate and check the validity of the numerical solutions.

Keywords: autoregressive models, eigenvalues space-time, least squares, multidimensional arrays, space-time forecasting

1. Introduction

Modern remote sensing technologies, for data acquisition and processing, provide large amounts of environmental data, with good coverage in space and time. When such data are in the form of sequences of digital images, properly georeferenced and equalized, then an entire timelapse video can be constructed. These movies allow dynamic monitoring and surveillance of earth areas subject to natural events (such as floods, landslides, and wildfires) and human interventions (such as urbanization, agriculture, and wars). A classical example is the Google Earth platform which edits videos from the imagery of LandSat and Copernicus satellites and broadcasts them through its YouTube channel. Recently, [1] has also implemented an online engine that enables users to build their videos at a global scale; it is continuously improved as regards space-time resolution and image quality.

Apart from descriptive and entertaining aspects, timelapse videos are useful for monitoring and surveillance purposes, to signal land hazards and risks. In this perspective, numerical methods for representing the video frames and for obtaining

meaningful information are crucial. From a statistical viewpoint, the video datasets are multidimensional (4D) arrays of space–time positive numbers; given their complexity, the application of dimensional reduction techniques, such as principal components (PC) and factor analysis, is necessary. Basic algebraic instruments are the eigenvectors of the covariance matrices and their projection properties in the space. These techniques are well known for single digital images; e.g., [2] apply PCA to image compression, [3] to object detection and image segmentation, and [4] to heterogeneous geodata layer fusion.

Point spatial data have a smaller size than images and allow for a formal treatment of the temporal component; e.g., [5] uses PC for directional (ridge) clustering of earthquake epicenters, [6] define a PCA approach in the attribute space that maintains the data structure in the spatiotemporal domain, and [7] develop space–time PCA toward functional statistical analysis. Operationally, [8] model networks of environmental stations as a multivariate AR system and use PC for reducing its dimension; they also study the effect of temporal ACR on PC extraction.

PCA of human videos has been considered in Ref. [9]; given the heterogeneity of scenes, the main goal is clustering the frames in homogeneous groups for subsequent uses (e.g., clip extraction). Liu et al. [10] used a nonlinear version of PCA to reach a more operational goal of automatic video editing. PCA methods for semantic video interpretation, to be applied in computer vision and robotics, have a long history mostly based on supervised classification; see Ref. [11] and reference therein. They require the construction of large and consistent datasets of annotated (human pre-classified) frames and sequences. Similarly, [12] using *neural* classifiers have tried to forecast video sequences out-of-sample, i.e., beyond the observed interval. This attempt is computationally demanding as it requires the calibration of complex neural networks, which are over-parameterized models from a statistical viewpoint.

In this paper, we consider timelapse videos of remote sensing data and use principal components both for synthesis and forecasting. In the time domain, PCs may resume video frames as long exposure photography, to have an instantaneous view of the land change. In the space domain, PCs may resume local series and reduce the dimension of space–time systems, to implement simpler factor models. We show that the presence of ACR is an issue from the theoretical viewpoint for PC estimation, but has limited practical effects both on data description and parameter estimates. We compare the forecasting performance, on out-of-sample frames, of factor AR models (that may be modeled as univariate time series) and space–time AR models (that are similar to multivariate systems; see [13]).

The paper is organized as follows: Section 2 deals with PCs in the time domain as a general tool of frame synthesis, compared to the simple arithmetic averaging. Section 3 discusses PCs in the spatial domain as a tool for building factor models; here, least squares (LS) estimator and forecasting algorithms are developed. Throughout, an extended numerical application to the Google Earth video of the Iquitos city (Peru) in the period 1984–2022 is carried out to illustrate and compare the methods.

2. Principal component analysis of videos

Remote sensing technologies and digital image processing generate numerical data on regular lattices. Typical datasets are in the form of 4D arrays of the type

$\mathbb{Y} = \{Y_{ijl}\}$, where $i = 1, 2 \dots n, j = 1, 2 \dots m$ are indices of pixel position (which may be transformed into latitude and longitude), $l = 1, 2 \dots k$ are the spectral bands ($k = 3$ for RGB color images), and $t = 1, 2 \dots T$ is the index of time (daily, monthly, and annual). The first approach of dimensional reduction is transforming the spectral bands into a single indicator, as the grayscale in the visible range, or normalized difference vegetation index (NDVI) in the near-infrared channel, to obtain 3D arrays. The resulting values Y_{ijt} are usually autocorrelated (in space and time) and non-stationary (with spatial and temporal trends).

The PC analysis of classical (2D) data matrices $\mathbf{X} = \{X_{ij}\}$, with N units and M variables, $N > M > 3$, is a technique of dimensional reduction to obtain a few linear combinations of the columns \mathbf{x}_j which capture most of the variability and allow visualization in 1-3D space. A direct application to image processing is to compress a color picture $\mathbb{X} = \{\mathbf{X}_l\}_1^3$ in its grayscale version, e.g., [2, 14]. The PCA technique vectorizes the RGB layers $\mathbf{x}_l = \text{vec}(\mathbf{X}_l)$, builds a $nm \times 3$ matrix $\mathcal{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3]$, estimates the first principal component $\mathbf{z}_1 = \sum_{l=1}^3 \nu_l \mathbf{x}_l$, and reshapes it as a new image \mathbf{Z}_1 . The questions now are: How can PCA be applied to a video $\mathbb{Y} = \{\mathbf{Y}_t\}_1^T$, with T frames, and what meaningful results may it produce?

Authors in Ref. [4] applied PCA to a 3D matrix with $k = 9$ geographic layers; the goal is to improve the GIS overlaying technique based on the arithmetic mean, which uses uniform weights $\nu_l = 1/k$. The fundamental step, before applying PCA, is rescaling the layers in the range $[0,1]$, with the transformation $X_{ijl} / \max_{ij}(X_{ijl})$; the PCA technique then provides non-uniform weights which enhance the most significant layers. Now, even in remote sensing sequences $\{\mathbf{Y}_t\}$ there is the goal of spatial mapping, but with the most ambitious purpose of representing the time-evolution of the phenomenon in a single frame (as in long-exposure photography). In this context, there are no problems of data heterogeneity, as the frames belong to the same band; rather, there is an issue of strong space-time ACR.

The steps for PCA investigation of a video are as follows: transform the initial 4D dataset in grayscale (frame by frame) or proceed separately on each color band. Vectorize the resulting 3D array $\mathbb{Y} = \{Y_{ijt}\}$, $n \times m \times T$, as

$$\mathcal{Y} = [\text{vec}(\mathbf{Y}_1), \text{vec}(\mathbf{Y}_2) \dots \text{vec}(\mathbf{Y}_t) \dots \text{vec}(\mathbf{Y}_T)], \quad (1)$$

obtaining a 2D matrix of size $nm \times T$. Consider its *centered* (mean zero) version

$$\mathcal{Y}_0 = (\mathcal{Y} - \mathbf{1}_{nm} \otimes \bar{\mathbf{y}}), \quad \bar{\mathbf{y}} = \mathbf{1}'_{nm} \mathcal{Y} / nm, \quad (2)$$

where $\mathbf{1}_{nm}$ is a unit vector of length nm . Compute the covariance matrix \mathbf{C} (which is symmetric and positive definite) and perform its spectral factorization

$$\mathbf{C} = \mathcal{Y}'_0 \mathcal{Y}_0 / nm, \quad \mathbf{C} = \mathbf{V} \Lambda \mathbf{V}', \quad (3)$$

where \mathbf{V}, Λ , are $T \times T$ matrices of eigenvectors and eigenvalues, where the latter are placed in decreasing order: $\lambda_k \geq \lambda_{k+1}$ within Λ .

Now, the meaningful first PCs of the space-time array \mathbb{Y} are given by

$$\mathbf{z}_0 = \mathcal{Y}_0 \mathbf{v}_1, \quad \text{projection on the first PC axis}, \quad (4)$$

$$\mathbf{z}_1 = \mathcal{Y} \mathbf{v}_1 / \|\mathbf{v}_1\|_1, \quad \text{weighted average of } \mathbf{Y}_t, \quad (5)$$

where \mathbf{v}_1 is the first eigenvector of the orthogonal matrix \mathbf{V} , and $\|\cdot\|_1$ is the absolute norm. Finally, the vectors $\mathbf{z}_0, \mathbf{z}_1$ must be reshaped as $n \times m$ matrices $\mathbf{Z}_0, \mathbf{Z}_1$ and encoded in uint8 format [0,255], to be represented and processed as images. While \mathbf{Z}_1 provides a weighted average of the frames, the image \mathbf{Z}_0 is more essential and may detect the major changes of the sequence $\{\mathbf{Y}_t\}$.

An issue in this approach, compared to the classical PCA, is the lack of independence of data. The space–time ACR of pixels may induce bias and inefficiency in the estimates; in particular, in the standard errors of the eigenvector \mathbf{v}_1 , see [15, 16]. As in regression models, a naïve method to improve the statistical properties is to include “lagged” terms into the system; in the above framework, this means computing the matrix \mathbf{V} in Eq. (3) on the *augmented* array

$$\mathcal{Y}^* = [\mathcal{Y}, \mathcal{Y}_1], \quad \mathcal{Y}_1 = \text{vec}(\mathbb{Y}_1), \quad \mathbb{Y}_1 = \{Y_{i-1,j-1,t-1}\}, \quad (6)$$

where the lagged array \mathbb{Y}_1 is integrated with missing terms, e.g., putting the column $\mathbf{y}_{m,t-1} = \mathbf{y}_{1,t-1}$. The resulting matrix (6) has size $nm \times T(T-1)$, and only the first T elements of \mathbf{v}_1^* are used for computing the PCA vectors $\mathbf{z}_0, \mathbf{z}_1$ in Eqs. (4) and (5).

As in time series, e.g., [8], a substantial reduction of ACR is provided by the space–time differencing $y_{ijt} = (Y_{ijt} - Y_{i-1,j-1,t-1})$. Since y_{ijt} also assume negative values, the nature of the implied coefficients \mathbf{v}_1 substantially changes, and they may not be suitable for the original data Y_{ijt} . Furthermore, reconstructing the target image \mathbf{Z} from the PCA image $\mathbf{z} = \{z_{ij}\}$ of the series y_{ijt} is difficult and biased. Indeed, this requires the spatial integration $Z_{ij} = Z_{i-1,j-1} + z_{ij}$, which in turn involves $n + m - 1$ initial values Z_{i1}, Z_{1j} ; these border values are arbitrary and may distort the entire image \mathbf{Z} .

Finally, for point (non-lattice) data, with matrices $\mathbf{X}_t = \{x_{skt}\}$, $N \times M$, equispaced in time t but irregularly distributed in space with coordinates (i_s, j_s) , [17] and [6] have considered a PCA approach based on a spatially weighted covariance matrix, as in the Moran index

$$\mathbf{C}_W = \frac{1}{NT} \sum_{t=1}^T (\mathbf{X}_t - \bar{\mathbf{x}})' \mathbf{W}_N (\mathbf{X}_t - \bar{\mathbf{x}}), \quad w_{ij} = \begin{cases} 0, & i = j, \\ 1, & \text{sparse,} \end{cases}$$

where \mathbf{W} is a $N \times N$ contiguity matrix of the points based on the assumption of interactions (e.g., nearest neighbors). The derivation of the matrix \mathbf{W} for lattice data is possible using geometrical rules of chess moves (e.g., rook, queen, etc.); in the presence of asymmetry, the positive definiteness of \mathbf{C} is preserved by

$$\mathbf{C}_W = \frac{1}{2nm} \mathcal{Y}'_0 (\mathbf{W}_{nm} + \mathbf{W}'_{nm}) \mathcal{Y}_0.$$

However, apart from the arbitrariness of the contiguity rules, for image data the building and use of the $nm \times nm$ array \mathbf{W} is numerically demanding for lattices [18], even in the lowest resolution case $(n, m) = (144, 256)$.

Anyway, the presence of ACR mostly affects the standard errors and test statistics of the estimates \mathbf{v}_1 (see [15] p. 299); hence, it may be a minor problem when using the PCs for image representation and processing. Instead, the mentioned corrections may introduce significant bias; thus, in the application, we mainly focus on Eqs. (4) and (5) for image synthesis.

2.1 Application to Google earth video

Google [1] creates timelapse videos based of LandSat and Copernicus satellite images, which are properly georeferenced and homogenized. The service is global, locally zoomable, and enables to evaluate how the Earth has changed over the past 40 years (since 1984). A set of high-resolution videos of interesting areas are put on the YouTube platform and can be downloaded; we consider the Iquitos city in Peru, located on the banks of Rio Amazonas, see [19]. As a consequence of the periodic floods, the change of the river bed between 1984 and 2022 is impressive, as well as the impact on urban growth (see **Figure 1**).

Given the computational load of the algorithms for a laptop computer with MATLAB software, we consider a low-resolution video (240p) and an area of about 18×22 Km. This yields a data array in black and white of size $n = 288$, $m = 368$, $T = 39$, which provides $N = 4,133,376$ observations. The color display is shown in **Figure 1**; notice the significant displacement of the river bed during 39 years.

Figure 2 provides the main statistics of the video in grayscale; the time-trends in Panel 2a show a rough 5-year cycle of the vegetation activity, which may be related to El Niño oscillation (ENSO). Panel 2b provides the correlation matrix of Eq. (3): $R = S^{-1/2}CS^{-1/2}$ with $S = C \odot I$ and shows that nearest frames are more correlated. Panel 2c shows the path of the first eigenvectors of Eqs. (3) and (6) and differences series y ; the first two are proportional (and coincide when are normalized by $1/\|v_1\|_1$), whereas the third is consistent with the first. This means that ACR has *not* a great effect on the estimates of v_1 . Finally, Panel 2d displays the relative eigenvalues $\lambda_k/\text{tr}(\Lambda)$ in the original and differenced series; it shows that PC1 is more significant on the original data, where it captures 71% of variability.

Finally, **Figure 3** displays in pseudocolor (with the MATLAB default colormap), the first PCA images from Eqs. (4) and (5). Panel 3a shows Z_1 , a weighted average of

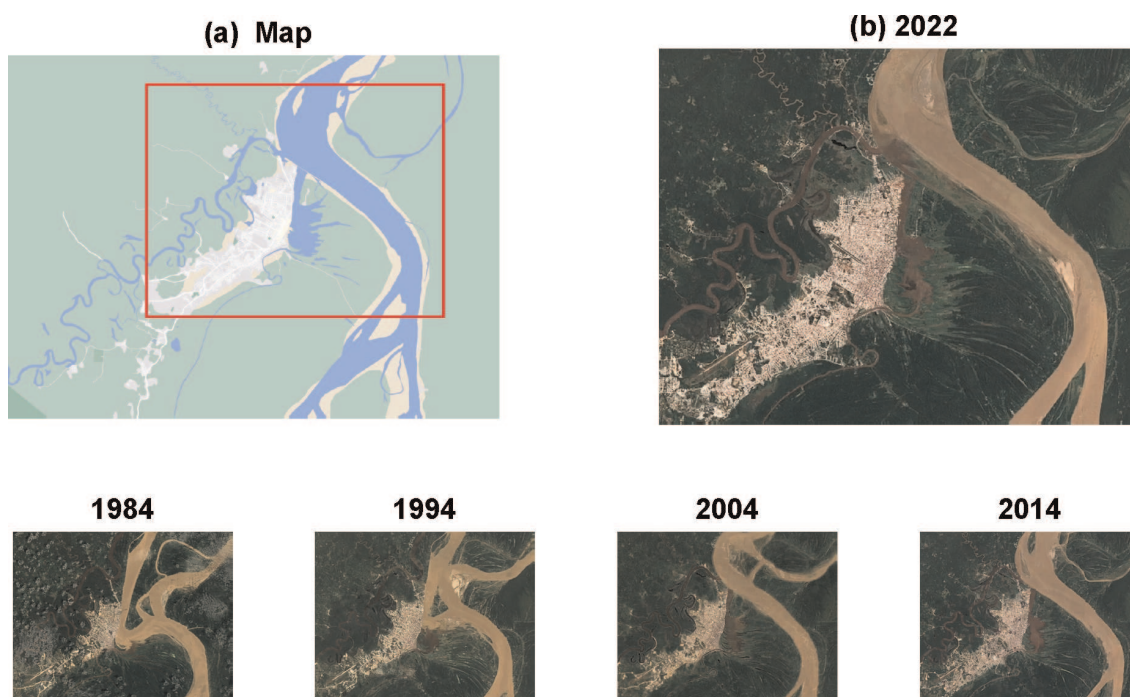


Figure 1. Google Earth video [19] of Iquitos (Peru) in the period 1984–2022: (a) Google map 2023; (b) LandSat color image 2022; (c) Low-resolution decadal frames.

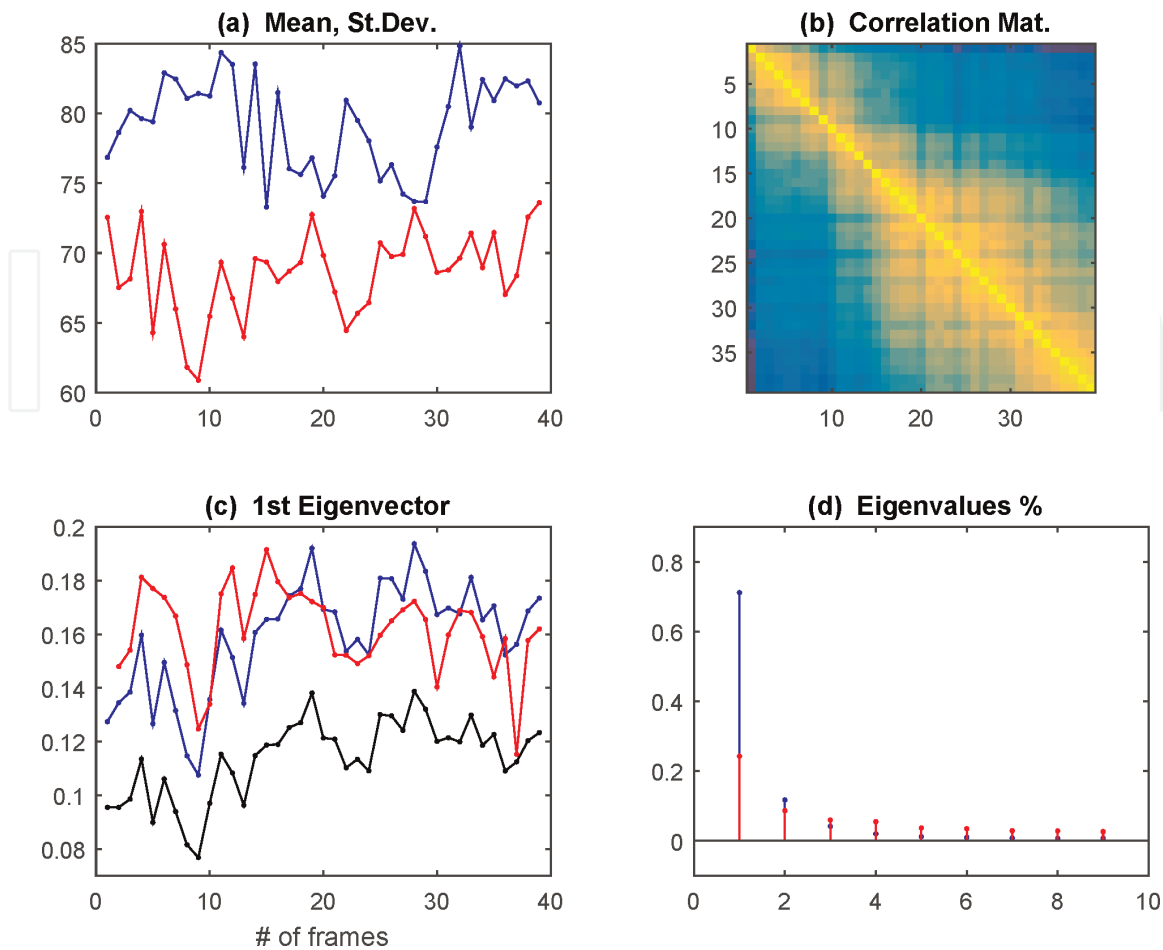


Figure 2. Statistics of the Iquitos video [19] in grayscale: (a) Trends of mean (blue) and standard deviation of the frames; (b) Correlation matrix of Eq. (3); (c) Path of PC1 eigenvectors \mathbf{v}_1 : original data (3) (blue), augmented data matrix (6) (black), and differenced series y (red); (d) Percent PC eigenvalues of the original and differenced (red) data.

the frames; its equalized version in Panel 3b enhances the land changes over 40 years. Panel 3c shows \mathbf{Z}_0 , the classical PC1; it is more effective in capturing the essential changes, placed where two basic colors (blue/yellow of forest/other) overlap. Panel 3d displays the PC1 obtained on the differenced series; it enhances the edges of the various images. The attempt to reconstruct the original image from the latter has provided meaningless results.

3. Dynamic factor models for videos

PCA is an explorative and descriptive technique of data analysis, whose numerical results may sometimes be difficult to interpret, also in image processing. Space-time ACR may be an issue for PCA estimates, but it is an asset for modeling Y_{ijt} and using the models for filtering and forecasting. In particular, out-of-sample forecasting of video frames is an objective test-bench for checking the effectiveness of numerical methods. In this context, PCA results may be useful for improving the parsimony of classical regression models, with the application of *dynamic factor* techniques (e.g., [20, 21]) to space-time data.

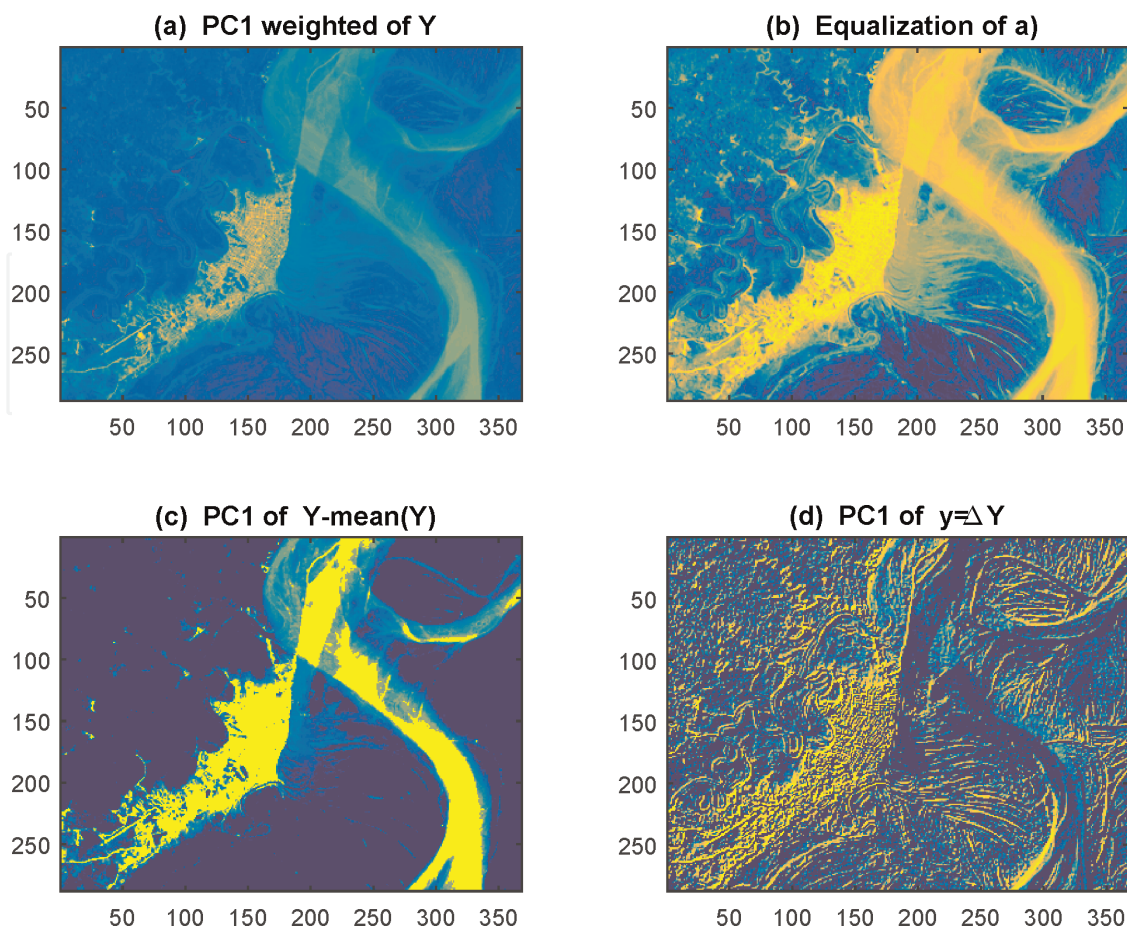


Figure 3. Pseudocolor display of the first PCA images of Eqs. (4), (5): (a) Weighted average Z_1 ; (b) Equalized version of Z_1 ; (c) Centered image Z_0 ; (d) Result on the differenced series.

3.1 Space-time systems

The space-time autoregressive (STAR) model puts each cell of the 3D array \mathbb{Y} in relation to the contiguous ones, such as $Y_{ijt} = g(Y_{i\pm h, j\pm k, t-l}) + e_{ijt}$, where $h, k, l = 1, 2 \dots p$ are spatial and temporal lags, $p > 0$ is the order of the dependence, and e_{ijt} is an unpredictable sequence. Under linearity of $g(\cdot)$, the STAR(p) representation can be denoted as

$$Y_{ijt} = \alpha_0 + \sum_{h=0}^p \sum_{k=0}^p \theta_{hk} Y_{i-h, j-k, t} + \sum_{l=1}^p \sum_{h=-p}^p \sum_{k=-p}^p \phi_{hkl} Y_{i-h, j-k, t-l} + e_{ijt}, \quad (7)$$

where $\theta_{00} = 0$ and $e_{ijt} \sim \text{IN}(0, \sigma_e^2)$ is an independent and normal (IN) sequence. The first part of the model (7), with parameters θ_{hk} , has a peculiar nature; it deals with the simultaneous relationships between the cells, and it is well studied in the (static) spatial AR literature, e.g., [18]. For reasons of identification and prediction, it has a *triangular* structure, so that filterings may proceed from the upper-left corner of every Y_t to the lower-right one in a sequential way.

Despite the simultaneous constraint, the number of parameters M_p of the model (7) is still large: for small $p = 2$ it becomes $M_p = 59$. A way to reduce the parametric complexity is to aggregate the pixels according to the geometric rules of contiguity of

the chess moves. By defining the notation $ij - 1$ when at least one of the indices i, j is lagged, then one may define the triangular (W) and ring (X) averages

$$W_{ij-1,t} = (Y_{i-1,j,t} + Y_{i,j-1,t} + Y_{i-1,j-1,t})/3, \quad (8)$$

$$X_{ij-1,t-1} = (Y_{i-1,j,t-1} + Y_{i+1,j,t-1} + Y_{i,j-1,t-1} + Y_{i,j+1,t-1} + \dots + Y_{i-1,j-1,t-1} + Y_{i+1,j-1,t-1} + Y_{i-1,j+1,t-1} + Y_{i+1,j+1,t-1})/8, \quad (9)$$

which lead to a constrained STAR (2) model with only 9 coefficients

$$Y_{ijt} = \alpha_0 + \theta_1 W_{ij-1,t} + \theta_2 W_{ij-2,t} + \phi_1 Y_{ij,t-1} + \phi_2 Y_{ij,t-2} + \dots \quad (10)$$

$$+ \beta_1 X_{ij-1,t-1} + \beta_2 X_{ij-2,t-1} + \beta_3 X_{ij-1,t-2} + \beta_4 X_{ij-2,t-2} + e_{ijt}, \quad (11)$$

The approach (8)–(11) of parametric reduction is subjective as depends on the aggregation rules; a more general solution arises by noting that all neighbors $\{Y_{i\pm h,j\pm k,t}\}$ of the series $\{Y_{ijt}\}$ can be “averaged” with the PCA technique and replaced by the first latent factor $\{Z_{ijt}\}$ as in Eqs. (4) and (5). Specifically, let $\mathbf{y}_{ij} = [Y_{ij1}, Y_{ij2} \dots Y_{ijT}]'$ be the time series located at ij , and consider its neighbors $\mathbf{y}_{i\pm h,j\pm k}$ in the square $\pm p$; then, for each ij one has the *temporal* PC1 component

$$\mathbf{z}_{ij} = \mathcal{Y}_{ij} \mathbf{v}_{ij1}, \quad \mathcal{Y}_{ij} = [\mathbf{y}_{i-p,j-p} \dots \mathbf{y}_{i-h,j+k} \dots \mathbf{y}_{i+p,j+p}], \quad (12)$$

where the data matrices \mathcal{Y}_{ij} have size $T \times (2p + 1)^2 - 1$.

Alternatively, with the $(2p + 1)^2 - 1$ spatially lagged arrays implied by Eq. (12), one may proceed in the space domain and extract the PCs at each t . Specifically, let $\mathbf{Y}_t = \{Y_{ijt}\}$ be the t -th frame and $\mathbf{Y}_{hk,t} = \{Y_{i+h,j+k,t}\}$, $h, k = 1 \dots p$ its spatially shifted companions, i.e., the frames built with all h, k -lagged neighbors of each site ij ; then, the *spatial* PC1 components are given by

$$\mathbf{z}_t = \mathcal{Y}_t \mathbf{v}_{t1}, \quad \mathcal{Y}_t = [\text{vec}(\mathbf{Y}_{-p,-p,t}) \dots \text{vec}(\mathbf{Y}_{-h,+k,t}) \dots \text{vec}(\mathbf{Y}_{p,p,t})], \quad (13)$$

where the data matrices \mathcal{Y}_t have size $nm \times (2p + 1)^2 - 1$, and reshaping \mathbf{z}_t provides a $n \times m$ PC1 layer for each t .

Both approaches (12) and (13) yield latent factor arrays \mathbb{Z} of size $n \times m \times T$, which may be used as explanatory (X) variables for the original data \mathbb{Y} , modeled as an ARX system. Further, the factor series may be represented by a simple AR scheme; this leads to the latent factor system of order $p = 2$ with 9 parameters, as Eqs. (10) and (11)

$$Z_{ijt} = \alpha_1 + \theta_1 Z_{ij,t-1} + \theta_2 Z_{ij,t-2} + u_{ijt}, \quad u_{ijt} \sim \text{IN}(0, \sigma_u^2), \quad (14)$$

$$Y_{ijt} = \alpha_2 + \phi_1 Y_{ij,t-1} + \phi_2 Y_{ij,t-2} + \beta_0 Z_{ijt} + \beta_1 Z_{ij,t-1} + \beta_2 Z_{ij,t-2} + e_{ijt}, \quad (15)$$

By means of the PCA framework (12) and (13), one can avoid to estimate the system (14) and (15) with the Kalman filter [21], which is nonlinear as regards factors and parameters. In the following, we compare the fitting and forecasting performances of the models (10), (11), (14), and (15); this requires the definition of estimation algorithms.

3.2 Statistical algorithms

If the extraction of latent factors involves linear matrix algebra (eigenvalue decomposition), the estimation of parameters of models (10), (11), (14), and (15) may be accomplished with least squares (LS). Rewrite the model (10) and (11) in regression form as follows

$$Y_{ijt} = \delta' \xi_{ijt} + e_{ijt}, \quad (16)$$

$$\delta' = [\alpha_0, \theta_1 \dots \phi_2, \beta_1 \dots \beta_4], \quad (17)$$

$$\xi'_{ijt} = [1, W_{ij-1,t} \dots Y_{ij,t-2}, X_{ij-1,t-1} \dots X_{ij-2,t-2}], \quad (18)$$

where the regressors W, X are generated from the data $\{Y_{ijt}\}$ with the formulas (8), (9). Thus, minimizing the sum $\sum_{ijt} e_{ijt}^2(\delta)$ provides the LS estimator

$$\hat{\delta}_N = \left(\sum_{t=p+1}^T \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} \xi_{ijt} \xi'_{ijt} \right)^{-1} \sum_{t=p+1}^T \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} \xi_{ijt} Y_{ijt}, \quad (19)$$

$$= \delta + \left(\sum_{t=p+1}^T \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} \xi_{ijt} \xi'_{ijt} \right)^{-1} \sum_{t=p+1}^T \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} \xi_{ijt} e_{ijt}, \quad (20)$$

where $p = 2$ is the order of the model (7). Eq. (20) is obtained from Eqs. (16), (19), and $N = (n - 2p)(m - 2p)(T - p)$ is the actual sample size, net of the borders.

The expression (20) shows the consistency of the estimator (19), when the contemporaneous terms W_{ijt} have a triangular structure as (8); in fact, it enables the sequential calculation of residuals and their independence from all regressors:

$E(\xi_{ijt} e_{ijt}) = \mathbf{0}$. Under this constraint and the conditions of stationarity and isotropy, one can show the classical convergence property [13].

$$\sqrt{N}(\hat{\delta}_N - \delta) \rightarrow N \left[\mathbf{0}, E(\xi_{ijt} \xi'_{ijt})^{-1} \sigma_e^2 \right], \quad \text{as } N \rightarrow \infty, \quad (21)$$

from which, the dispersion matrix of the estimator (19), (20) is given by

$$\hat{\Sigma}_{\delta} = \mathbf{R}_N^{-1} \hat{\sigma}_e^2, \quad \mathbf{R}_N = \sum_{t=p+1}^T \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} \xi_{ijt} \xi'_{ijt}, \quad (22)$$

$$\hat{\sigma}_e^2 = (N - M_p)^{-1} \sum_{t=p+1}^T \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} (Y_{ijt} - \hat{\delta}'_N \xi_{ijt})^2, \quad (23)$$

where $M_p = 9$ is the length of δ when $p = 2$. Unlike maximum likelihood, the algorithm (19) can manage datasets of large dimensions; i.e., with high pixel resolution $n \times m$ and high frequency frames T . Also the inversion of the $M_p \times M_p$ matrix (22) generally does not involve numerical issues. The formulas (19)–(23) can also be applied to the factor model (14), (15), in an even simpler way.

Forecasting. As regards prediction, given the linearity of Eqs. (10), (11) and (14), (15) the forecasts can be obtained with the chain rule of forecasting; i.e., by “pushing forward” the entries of the vector ξ_{ijt} and updating it with the past forecasts:

$$\hat{Y}_{ij,T+l} = E(Y_{ij,T+l} | \mathbf{Y}_{T-t}, t \geq 0) = \delta' \hat{\xi}_{ij,T+l}, \quad (24)$$

$$\hat{\xi}_{ij,T+l} = [1, \hat{W}_{ij-1,T+l} \dots \hat{Y}_{ij,T+l-2} \dots \hat{X}_{ij-2,T+l-2}]', \quad l = 1, 2 \dots L, \quad (25)$$

where $\hat{X}_{ij-2,T+l-2} = (\hat{Y}_{i-2j-2,T+l-2} + \hat{Y}_{i-2j-1,T+l-2} + \dots + \hat{Y}_{i+2j-2,T+l-2})/16$ are the lagged ring averages for $l > 2$, and so on.

A computational issue with the formula (24) and (25) applied to the model (10) and (11) arises from the presence of the contemporaneous terms $\hat{W}_{ij-k,T+l}, k = 1, 2$, which depend on the forecasts $\hat{Y}_{i-k,j-k,T+l}$ themselves. Although these elements satisfy the unilateral constraint, they require the upper-left border values $\hat{Y}_{ij,T+l}, i, j \leq p$ to start. These quantities can be obtained as forecasts of AR models applied to the marginal cells, or by simply setting $\hat{Y}_{ij,T+l} = Y_{ij,T}$, as in random-walk models. Hence, by the concatenation of forecasts in Eqs. (24) and (25), these marginal values influence the entire forecast frame $\hat{\mathbf{Y}}_{T+l}$. This effect can be checked empirically with out-of-sample forecasting, i.e., predicting data that have been omitted from parameter estimation. The typical statistics used for evaluation are the mean squared forecast errors (MSFE) and its relative R^2 index

$$\text{MSFE}_{T'}(l) = \frac{1}{(n-2p)(m-2p)} \sum_{i=p+1}^{n-p} \sum_{j=p+1}^{m-p} (Y_{ij,T'+l} - \hat{Y}_{ij,T'+l})^2, \quad (26)$$

$$R^2(l) = 1 - \text{MSFE}_{T'+l} / \hat{\sigma}_{\mathbf{Y}_{T'+l}}^2, \quad l = 1, 2 \dots L, \quad (27)$$

where $T' = T - L$. These statistics enable to compare the performance of the models (10), (11) and (14), (15); the best one is that with lowest (26). The index (27) provides a measure of the reliability of forecasts in decision-making.

Simulations. To test the statistical properties of the LS estimator (19), we perform simulation experiments on the model (10), (11) with stable/unstable parameters and unilateral/multilateral simultaneous components. In particular, the multilateral design of the term W follows the cross (rook) scheme

$$W_{ij-1,t}^* = (Y_{ij-1,t} + Y_{ij+1,t} + Y_{i-1,j,t} + Y_{i+1,j,t})/4,$$

in this case, the LS method should be biased for all parameters. We perform 500 replications of the system (10), (11) with $p = 1, n = 13, m = 12, T = 11$, and $|\delta_i| = 0.5, 1.0$, for $i = 1 \dots 4$; notice that the actual sample size $N = (n-2)(m-2)(T-1) = 1100$ is large enough. Simple means, root mean squared errors ($\text{RMSE} = [\text{Var} + \text{Bias}^2]^{1/2}$) and mean P -value of the normality test are reported in **Table 1**.

Since the parameters have the same size, their statistics can be averaged to provide synthetic indicators. As a result, one may note that the worst performance is that of multilateral W with “unstable” coefficients; whereas the model with unilateral W benefits from the greater signal-to-noise ratio σ_Y / σ_e induced by $|\delta_i| = 1$. This property is known as super-consistency [13], but its side-effect is the non-normality of estimates; this complicates statistical inference, requiring non-standard distributions as in the tests for unit-roots of time series.

Param.	Value	W-triangular			W-cross		
		Mean	RMSE	N-test	Mean	RMSE	N-test
α_0	0.5	0.501	0.031	0.414	0.211	0.291	0.055
θ_1	-0.5	-0.498	0.019	0.111	-0.433	0.072	> 0.5
ϕ_1	0.5	0.500	0.007	0.001	0.755	0.260	> 0.5
β_1	-0.5	-0.501	0.016	0.002	0.051	0.556	> 0.5
Abs. Ave.	0.5	0.500	0.018	0.132	0.363	0.295	0.389
α_0	1.0	1.001	0.030	> 0.5	-9.593	13.56	> 0.5
θ_1	-1.0	-1.000	0.000	0.001	-2.401	1.409	0.011
ϕ_1	1.0	1.000	0.000	0.001	1.086	0.107	0.001
β_1	-1.0	-1.000	0.000	0.001	2.707	3.709	0.001
Abs. Ave.	1.0	1.000	0.008	0.126	3.946	4.697	0.128

Table 1. Performance of the LS estimator (19), applied to the model (10), (11) with order $p=1$, with triangular and cross (rook) contemporaneous component W , and stable (0.5) and unstable (1.0) coefficients; mean, RMSE, mean P -value are over 500 replications.

3.3 The empirical application

We illustrate and check the methods discussed so far on the same dataset as Section 2, displayed in **Figure 1**. The first step in model building is the extraction of the PC1 from the array \mathbb{Y} with the approaches (12) and (13). With $p = 2$, we select 24 neighbor series of each site ij ; the results are displayed in **Figure 4**. Panel (a) shows that the PC1 explains on average about 85% of the frame variability, with a growing trend; further, Panel (b) shows its spatial pattern, with a greater quote in areas subject to major changes. Panels (c,d) show that the PC1 arrays have an image representation that is sharp for the method (12) and blurred for (13).

Apart from the descriptive aspects, PC1 data are necessary for modeling and prediction; **Table 2** provides the parameter estimates of the models (10), (11), (14), (15) with the LS method (19). The estimations are performed on the frames $[\mathbf{Y}_3 \dots \mathbf{Y}_{T-2}]$, where $\mathbf{Y}_1, \mathbf{Y}_2$ are starting values and $\mathbf{Y}_{T-1}, \mathbf{Y}_T$ serve for forecasting evaluation. The results are very significant in terms of t -type statistics because these are inflated by the large sample size $N = 3,618,160$; the evaluation of coefficients should then be based on their size, e.g., $|\hat{\delta}_i| > 0.1$ as they are related to ACR. In any case, the R^2 indices of goodness of fit are very satisfactory, and, as regards the model (10), (11), the simultaneous components W have a leading role. However, the best fitting model is Eqs. (14) and (15) with factor component Z estimated as in Eq. (13), and normalized by $1/\|\mathbf{v}_1\|$; in the following, we evaluate their forecasting ability.

Forecasting. The prediction ability of the models of **Table 2** is evaluated on the last two frames $\mathbf{Y}_{T-1}, \mathbf{Y}_T$, which were kept out of parameter estimations. The forecasts were computed with the function (24), (25) and then evaluated with the statistic (26), (27), with starting point $T' = 2020$; in these computations, the forecasts were expressed in uint8 format, and the results are displayed in **Table 3**. As for the in-sample fitting, the best model is (14), (15) with factor (13) normalized by $1/\|\mathbf{v}_1\|$; it is

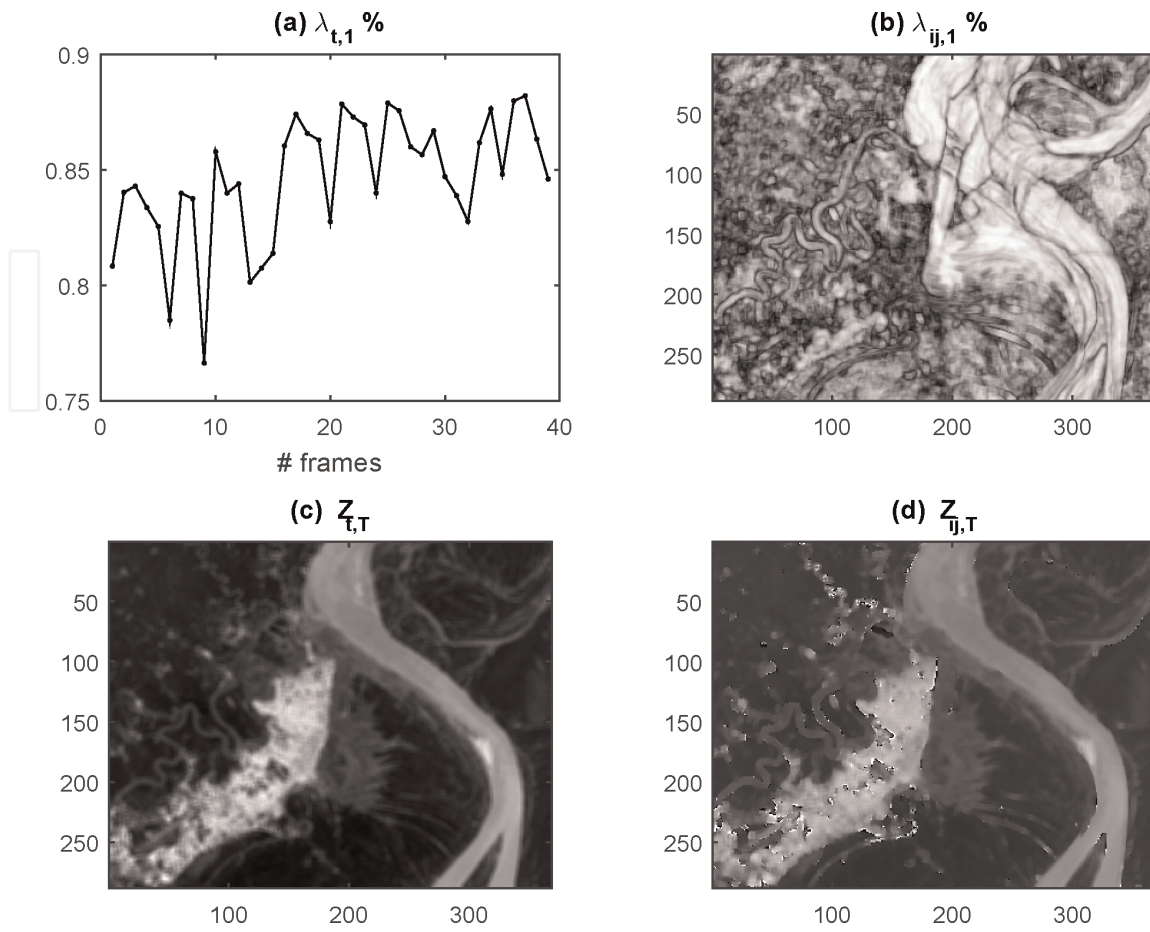


Figure 4. Results of the extraction of the PC1 from the 24 neighbors of each ij : (a,c) Spatial direction (13); (b,d) Temporal direction (12); (a,b) Quote of variance explained by PC1; (c,d) Images of PC1 at time $T = 2022$.

Param.	Model (10)	(10) without W	Param.	Model (12), (15)	Model (13), (15)
α_0	0.804 (62.4)	2.958 (150.5)	α_1	3.284 (225.3)	4.768 (268.6)
θ_1	0.914 (1580)	.	θ_1	0.641 (1328)	0.651 (1330)
θ_2	-0.197 (-338.2)	.	θ_2	0.319 (662.)	0.293 (599.7)
ϕ_1	0.494 (641.1)	0.526 (445.5)	α_2	-0.436 (-36.6)	1.433 (102.3)
ϕ_2	0.204 (267.1)	0.217 (185.5)	ϕ_1	0.5252 (1048)	0.577 (1163)
β_1	-0.450 (-288.0)	0.019 (8.18)	ϕ_1	0.212 (426.9)	0.248 (503.5)
β_2	0.139 (117.0)	0.099 (56.4)	β_0	1.067 (2536)	0.722 (1812)
β_3	-0.185 (-121.2)	-0.006 (-2.53)	β_1	-0.565 (-790.5)	-0.408 (-701.1)
β_4	0.071 (62.5)	0.110 (63.2)	β_2	-0.233 (-347.3)	-0.159 (-298.1)
σ_e^2	90.39	212.05	σ_e^2	77.60	115.07
R^2	0.932	0.842	R^2	0.942	0.914

Table 2. LS estimates (and t -type statistics) (19), of the parameters of the model (10), (11) (with and without the contemporaneous components W); and model (14), (15) (with factors (13) and (12) expressed in uint8 format) on the data of Figure 1 in the period 1986–2020.

Year	(10)+W	(10)-W	(12), (15)	(13), (15)
2021	248.75	170.05	166.47	167.81
	(0.850)	(0.898)	(0.899)	(0.898)
2022	428.75	288.27	281.74	283.93
	(0.756)	(0.836)	(0.838)	(0.837)

Table 3. MSFE statistics (17), and R^2 indexes (in parenthesis), of the AR models in Table 2 on the images 2021, 2022 of the Iquitos video; Bold indicates the best result.

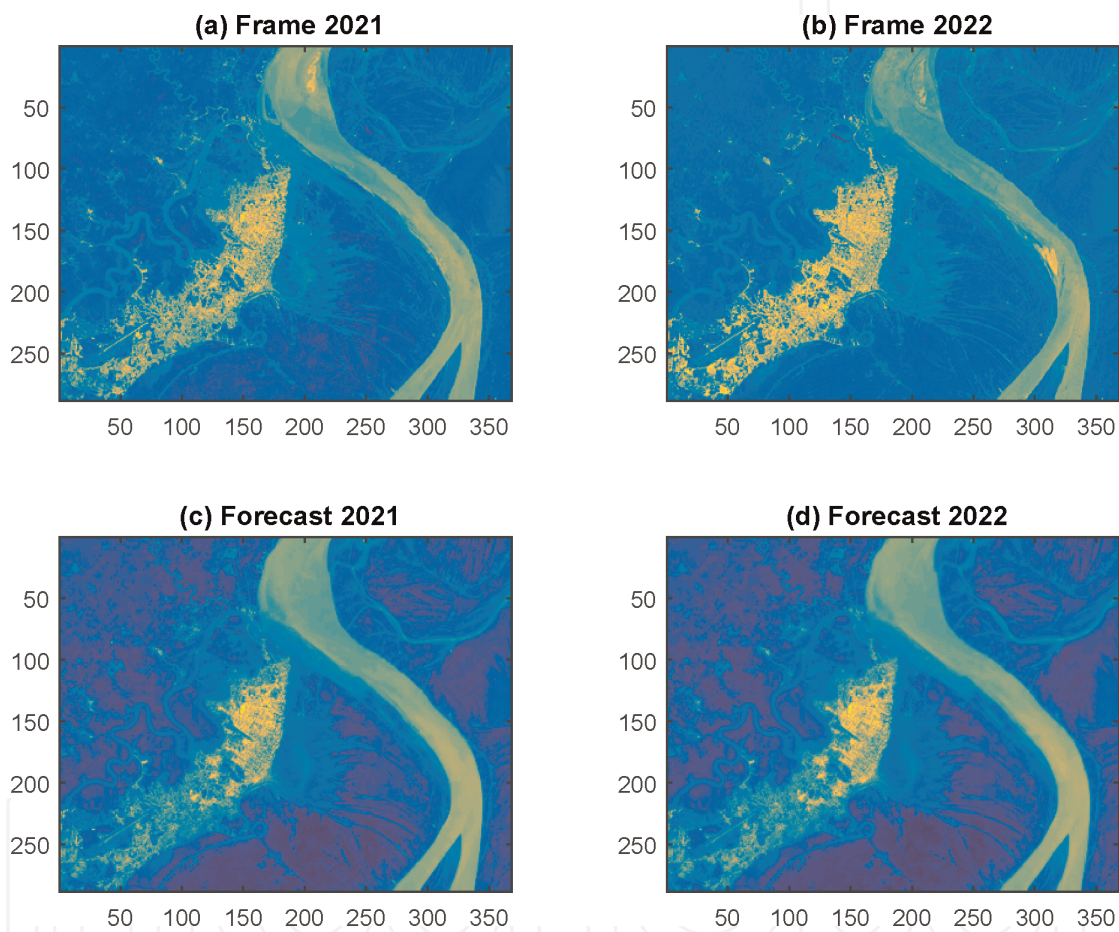


Figure 5. Comparison of real frames and best forecasts in Table 3, obtained with algorithm (24), (25): (a,b) Real images 2021, 2022; (c,d) Forecasts of the model (13)–(15).

slightly better than model (14), (15) with factor (12) and model (10), (11) without the simultaneous components. In general, while the W components improve fitting, they may affect forecasts; the reason is partly due to the starting values in the (upper-left) borders, which are established with the random walk rule, e.g., $\hat{Y}_{11,T+1} = Y_{11,T}$.

Finally, Figure 5 displays the best forecasts for 2021, 2022 (with smallest MSFE in Table 3) and compares them with the ground images (in pseudocolor MATLAB). The spatial paths of forecasts are consistent with the actual images, although they show fewer details in the urban area; further, the $R^2 = 0.9$ coefficient in Table 3 confirms the reliability of these forecasts.

4. Conclusions

This paper is concerned with the application of multivariate statistical methods to timelapse videos of remotely sensed data. Such movies are nowadays available on Internet for entertaining and scientific purposes, and their modeling is challenging both for the size of datasets (big data) and the complexity of the phenomena they represent. Classical multivariate techniques of data reduction, such as principal components, are useful both for reasons of data description and model building. In particular, PCA may condense the frames as a long exposure photography (see **Figure 3**) and provide local components for dynamic factor models, as Eqs. (14), (15). These are smart alternatives to complex space–time AR models.

The PCA of videos in the time domain provides two basic solutions for frame fusion, depending on whether it considers a weighted average of original images or the centered data array, see Eqs. (4), (5). In the second case, a uint8 transformation of the estimates is necessary to appreciate the result as an image; this transformation, by censoring negative values, makes the final result more essential and highlights major land changes (see **Figure 3c**). However, also factor models are useful for descriptive purposes as highlights the spectral properties of data arrays in the time domain and the spatial domain (see **Figure 4a,b**).

Mathematical modeling is useful for out-of-sample forecasting; this, in turn, is useful for monitoring and surveillance. The paper has provided a factor model framework which is more parsimonious and effective than the classical STAR systems. In prediction, there is usually a negative trade-off between model complexity (which improves fitting) and out-of-sample performance. While the model (7) requires ad-hoc aggregations of neighboring pixels, as in Eqs. (8), (9), in the model (14), (15) one has only to select the time order. As regards the spatial dimension, one may increase the span p in Eqs. (12) and (13) without increasing the number of coefficients of (14), (15); the only drawback is PC estimation at the borders of the lattice which requires symmetrical or circulant integrations.

A comparison of the models in **Table 3** with $p = 2$ shows that there is *not* significant difference in the performance of factor models and simplified STAR (without the simultaneous component W). In particular, all R^2 coefficients are close to 90%; however, the distance between the two modelings may increase as p .

Data and software

They are available at the site: <https://it.mathworks.com/matlabcentral/fileexchange/173895-pca-and-factor-ar-models-for-timelapse-video-data>

IntechOpen


IntechOpen

Author details

Carlo Grillenzoni
IUAV University, Venice, Italy

*Address all correspondence to: carlog@iuav.it

IntechOpen

© 2024 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Google Earth. Google Earth Engine. Mountain View (CA): Google Earth; 2024. Available from: <https://earthengine.google.com/timelapse> [Accessed: June 30, 2024]
- [2] Mudrová M, Procházka A. Principal Component Analysis in Image Processing. 2005. Available from: <https://www2.humusoft.cz/www/papers/tcp05/mudrova.pdf> [Accessed: June 30, 2024]
- [3] Pandey PK, Singh Y, Tripathi S. Image processing using principle component analysis. *International Journal of Computer Applications*. 2011; **15**(4):37-40
- [4] Salata S, Grillenzoni C. A spatial evaluation of multifunctional ecosystem service networks using principal component analysis: A case of study in Turin, Italy. *Ecological Indicators*. 2021; **127**:107758
- [5] Grillenzoni C. Sequential mean shift algorithms for space–time point data. *Environmental Earth Sciences*. 2018; **77**: 336
- [6] Stahlschmidt S, Härdle WK, Thome H. An application of principal component analysis on multivariate time-stationary spatio-temporal data. *Spatial Economic Analysis*. 2015; **10**(2): 160-180
- [7] Krzyśko M, Nijkamp P, Ratajczak W, Wolyński W, Wenerska B. Spatio-temporal principal component analysis. *Spatial Economic Analysis*. 2024; **19**(1): 8-29
- [8] Zamprogno B, Reisen VA, Bondon P, Aranda-Cotta HHC, Reis NC. Principal component analysis with autocorrelated data. *Journal of Statistical Computation and Simulation*. 2020; **90**(12):2117-2135
- [9] Sahouria E, Zakhor A. Content analysis of video using principal components. *IEEE Transactions on Circuits and Systems for Video Technology*. 1999; **9**(8):1290-1298
- [10] Liu Y, Liu Y, Chan KCC. Dimensionality reduction for descriptor generation in rushes editing. In: *IEEE International Conference on Semantic Computing, Santa Monica (CA)*. New York (NY); 2008. pp. 104-111
- [11] Yousif AJ, Al-Jammas MH. Exploring deep learning approaches for video captioning: A comprehensive review. *e-Prime*. 2023; **6**:100372
- [12] Weissenborn D, Täckström O, Uszkoreit J. Scaling autoregressive video models. In: *International Conference on Learning Representations (ICLR 2020)*. ICLR 2020 Conference. 2020. Available from: <https://iclr.cc/Conferences/2020>
- [13] Grillenzoni C. Adaptive spatio-temporal models for satellite ecological data. *Journal of Agricultural, Biological and Environmental Statistics*. 2004; **9**: 158-180
- [14] Chang R. Application of principal component analysis in image signal processing. In: *Proceedings International Conference on Image, Signal Processing and Pattern Recognition (ISPP 2022)*. Vol. 12247. SPIE Digital Library; 2022. Available from: <https://2022.icispp.com/>
- [15] Jolliffe IT. *Principal Component Analysis*. New York, NY: Springer; 2002
- [16] Vanhatalo E, Kulahci M. Impact of autocorrelation on principal components

and their use in statistical process control. *Quality and Reliability Engineering International*. 2015;**32**(4): 1483-1500

[17] Jombart T, Devillard S, Dufour AB, Pontier D. Revealing cryptic spatial patterns in genetic variability by a new multivariate method. *Heredity*. 2008; **101**(1):92-103

[18] Grillenzoni C. Forecasting lattice and point spatial data: Comparison of unilateral and multilateral SAR models. *Forecast*. 2024;**6**(3):700-717

[19] Google Earth. Iquitos, Peru - Earth Timelapse. Mountain View (CA): Google Earth; 2023. Available from: <https://www.youtube.com/watch?v=ZHhByopdLY4> [Accessed: June 30, 2024]

[20] Stock JH, Watson MW. Dynamic factor models, factor-augmented vector autoregressions, and structural vector autoregressions in macroeconomics. In: *Handbook of Macroeconomics*. Vol. 2. Amsterdam: Elsevier; 2016. pp. 415-525

[21] Krantz S. *Dynamic Factor Models: A Very Short Introduction*. Cran. R Project. Virtual Publisher; 2023. DOI: 10.32614/CRAN.package.dfms. Available from: <https://cran.r-project.org/>