

Higher-order Autoregressive Models for Dynamic Textures

Midori Hyndman, Allan Jepson, David Fleet
Department of Computer Science
University of Toronto
mhyndman, jepson, fleet@cs.toronto.edu

Abstract

Dynamic textured sequences are characterized by the interactions between many particles or objects in the scene. Based on earlier work the images of the sequence are interpreted as the output of a linear autoregressive process driven by white Gaussian noise. We extend earlier work by increasing the amount temporal information included when learning the motion in the scene, allowing the models to capture complex motion patterns which extend over multiple frames, thereby increasing the perceptual accuracy of the synthesized results. To overcome problems of dynamic model stability, we apply Burg's Maximum Entropy Spectral Analysis technique for parameter estimation, which is found to be reliably stable on smaller samples of training data, even with higher-order dynamics.

1 Introduction

A dynamic texture is an image sequence characterized by the interactions between many particles or objects in the scene. Examples of dynamic textures include, flames flickering, leaves blowing, and crowds observed from a distance. For such scenes, learning the motion by segmenting and tracking the trajectory of each component is computationally intensive; a holistic representation of the scene and the motion is motivated.

One well-known approach is to infer linear, autoregressive models of dynamic textures. The frames of the image sequence are interpreted as the output of stochastic process driven by white Gaussian noise. The appearance of the scene is described by a subspace model and the dynamics of the scene are captured within this subspace by a generative model that determines the hidden state of the system. Previous work using autoregressive models for dynamic texture synthesis, [6] in particular, used a first-order dynamical model. Incorporating only information from the preceding state prevents the capture of oscillations and other motions that rely on higher-order temporal dependencies in the image sequence. Also, with first-order models the perceptual quality of the synthesized scene deteriorates within a short interval.

In this paper, we propose the use of higher-order autoregressive dynamic texture models. We find that increasing the amount of temporal information when learning the interframe dependencies allows the model to capture complex patterns which extend over multiple frames, increasing the perceptual accuracy of the synthesized results.

When incorporating a higher-order dynamical model, issues of model stability arise. To overcome these issues we apply the Maximum Entropy Spectral Analysis (MESA)

technique for linear prediction [3]. This approach is common in control theory but, to our knowledge, not typically used in the field of computer vision and new to dynamic texture modeling. This estimation technique is more reliably stable and perceptually accurate on smaller samples of training data, even with a higher-order dynamical model, than when using the Yule-Walker equations.

2 Related Work

Texture analysis and synthesis was pioneered by Julesz [12] with the observation of the correlation between statistical and perceptual similarity of textured images. Since then, many image-based rendering techniques have emerged for synthesizing static and dynamic textured scenes.

Non-parametric methods synthesize images using probabilistic sampling of the observed data, either pixel by pixel [5, 9, 10, 27] or by copying patches [8, 15, 28]. In non-parametric dynamic texture synthesis, notable results have emerged using patch-based techniques, where image patches are interpreted as segments of the image sequence [14, 22]. The synthesized temporal textures generated with these methods tend to be perceptually realistic, however, the images are limited to samples of the original sequence. Moreover, because a model of the scene is not explicitly inferred, the synthesized results cannot be generalized and further processing, viz. classification [4], is limited.

Parametric methods for dynamic textures were introduced in [18]. Modeling dynamic textures as the output of a spatio-temporal autoregressive process was shown to be successful with certain classes of textures and motions [25], however, the framework could not model spatially non-stationary motion, such as rotation. In [6], these limitations are addressed by representing dynamic textures as the output of a first-order subspace process with a Gaussian driving distribution,

$$y_t = Cx_t \quad (1)$$

$$x_t = -Ax_{t-1} + Wv_t, \quad v_t \sim \mathcal{N}(0, I). \quad (2)$$

In their appearance model (1) each image y_t is considered an expansion of the state variable x_t which is defined in the principal component subspace. In their dynamic model (2) the current hidden state of the system is derived from a linear combination of the elements in the preceding state, described by matrix A , and additive Gaussian noise with covariance WW^T is used to stochastically drive the process.

In [6], the dynamic model parameters are learned within the appearance model subspace. If the dynamic and appearance information is non-separable, this approach determines only an approximation to the optimal parameter estimates. To guarantee an optimal parameter estimate, the appearance and dynamic model parameters would be learned simultaneously. In [26] the dynamic model is computed in the original input space and the appearance model is constructed to retain a maximum amount of the information with respect to the dynamics of the system. In [23] an iterative approach is suggested where the results of [26] are used for initialization. Unfortunately these techniques are computationally infeasible on common workstations, given high-dimension input such as image data. Instead, we implement a closed-form solution to approximate the optimal parameter estimates, as in [6].

In contrast to the prevalent use of first-order dynamical models in earlier work, we advocate the use of higher-order models in the autoregressive process. We show that

higher-order models produce improved synthesized sequences with perceptual quality maintained over a longer time interval. The advantages of the autoregressive framework are preserved: separating the appearance and dynamical components enables classification [4], facilitates recognition applications [21] and provides a more manipulable model to explore video editing [7]. Moreover, incorporating influence from states lagged further in time captures the temporal dependencies that are capable of modeling oscillations. Using higher-order dynamics, however, introduces issues of model stability. We draw on a parameter estimation technique used in control theory to improve the stability of the resulting model, the Maximum Entropy Spectral Analysis technique [3].

3 Autoregressive Model

In this work a dynamic texture is modeled as the output of an autoregressive process consisting of an appearance model, which determines the state of the system, and a dynamic model, which captures how the states change over time:

$$y_t = Cx_t + u_t, \quad u_t \sim \mathcal{N}(0, B), \quad (3)$$

$$x_t = -\sum_{i=1}^{\mu} F_{\mu,i}x_{t-i} + Wv_t, \quad v_t \sim \mathcal{N}(0, I). \quad (4)$$

At time t , each image y_t , in column vector form, is defined by the expansion of a hidden state variable, x_t . In the generative appearance model (3) the matrix C projects the subspace representation into the image space, and the zero-mean normally distributed additive noise captures the uncertainty with covariance B . The dynamic model (4) contains a deterministic component (i.e. a Markov-model described by $F = \{F_{\mu,1}, F_{\mu,2}, \dots, F_{\mu,\mu}\}$) and a stochastic component (i.e. a Gaussian driving distribution with covariance WW^T). As in [6], we ignore the additive appearance noise u_t (i.e., take $u_t \equiv 0$) and capture all additive process noise within the driving distribution v_t .

We learn the parameters C , F , and W for the μ -order autoregressive model of an image sequence. Initializing the model with a set of μ consecutive image frames, one can generate novel image sequences which resemble the original data. The model is successful if the synthetic sequences are perceptually similar to the original sequence and, ideally, the model parameters are sufficiently generalizable to support recognition tasks [21].

3.1 Appearance Model

While the optimal estimator finds C , F , and W simultaneously, following [6], we use principal component analysis (PCA) to define the appearance model parameters C and we learn the dynamical model parameters F and W within the PCA subspace. To determine C , each image of the observed sequence is converted into column vector form, the mean image is subtracted, and the resulting vectors are concatenated to form Y_1^τ , a matrix of size $p \times \tau$ where p is the number of pixels per image times the number of colour channels, and τ is the number of images ($\tau < p$). Let $Y_1^\tau \equiv U\Sigma V^T$ be a singular value decomposition (SVD) where U is $p \times p$, Σ is $p \times \tau$, and V is $\tau \times \tau$. We choose $q \ll p$ and define $C \equiv \hat{U}$ where \hat{U} is a matrix containing the first q principal directions found in the columns of U . Let \hat{V} be the first q columns of V and $\hat{\Sigma}$ be a diagonal matrix of the q largest singular values from Σ . We define the subspace representation of Y_1^τ to be $X_1^\tau \equiv \hat{\Sigma}\hat{V}^T$. There are non-linear alternatives which, in future work, could be used within the appearance model; in particular, [20] is developed specifically for spatial textures.

3.2 Dynamic Model

The dynamic model comprises a deterministic linear model and a Gaussian driving distribution. The true dynamical process which generated the original sequence may contain both linear and non-linear components. Nonetheless, we assume that a linear autoregressive model is sufficient to describe the visual process. Information not captured within the linear component is modeled in the stochastic component of the dynamics.

The Yule-Walker equations can be used to solve for the coefficients of the dynamic model in the least squares sense, as in [6]. However, this approach assumes the stationarity of the training data sample statistics, an assumption which breaks down for short dynamic texture segments. As the order of the dynamic model increases and accuracy of the sample statistics deteriorate, the dynamic model determined with the Yule-Walker method is often unstable. In an unstable linear system, the predicted states tend towards infinity over time, resulting in perceptually unrealistic synthesized sequences.

The Maximum Entropy Spectral Analysis (MESA) technique was developed for single channel signals [3] and extended to handle multidimensional data [16, 24]. Although common in the control theory literature, to our knowledge this technique has not been applied to dynamic textures. When modeling dynamic textures, in practice only small portions of the sequences are available. Inaccurate models result when the higher-order sample statistics do not adequately reflect the structure in data. The main contribution of MESA is that by using a recursive approach the higher-order autocorrelations are never calculated directly from the sample data, despite the assumption of stationarity. An additional advantage of MESA, is that the stability of the resulting model is guaranteed [13]. Moreover, compared to using the Yule-Walker equations, we found that fewer training frames are necessary to obtain an accurate model [11].

MESA uses a recursive approach that depends on the coefficients of both forward and backward models,

$$x_t = - \sum_{i=1}^{\mu} F_{\mu,i} x_{t-i} + e_{\mu,t}, \quad (5)$$

$$x_t = - \sum_{i=1}^{\mu} B_{\mu,i} x_{t+i} + b_{\mu,t}. \quad (6)$$

where $e_{\mu,t}$ and $b_{\mu,t}$ are the forward and backward residuals. In (5), future states are predicted using the past states of the system, whereas in (6) past states are predicted using future data. The coefficients of an μ -order model are as follows,

$$\mathbf{F}_{\mu} = [I \ F_{\mu,1} \ F_{\mu,2} \ \dots \ F_{\mu,\mu}]^T, \quad (7)$$

$$\mathbf{B}_{\mu} = [B_{\mu,\mu} \ \dots \ B_{\mu,2} \ B_{\mu,1} \ I]^T, \quad (8)$$

where I is an identity matrix of size $q \times q$. These model coefficients have the following recursive relationship [3],

$$\mathbf{F}_{\mu} = \begin{bmatrix} \mathbf{F}^{\mu-1} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{B}^{\mu-1} \end{bmatrix} F_{\mu,\mu}, \quad (9)$$

$$\mathbf{B}_{\mu} = \begin{bmatrix} \mathbf{F}^{\mu-1} \\ 0 \end{bmatrix} B_{\mu,\mu} + \begin{bmatrix} 0 \\ \mathbf{B}^{\mu-1} \end{bmatrix}. \quad (10)$$

Matrices $F_{\mu,\mu}$ and $B_{\mu,\mu}$ are called the *reflection coefficients* and 0 is the zero matrix; all are of size $q \times q$. To solve for \mathbf{F}_{μ} in (9), we solve for the reflection coefficients in a

least squares sense, minimizing the squared residual error averaged over the sequence. The expected value of the reflection coefficients given the forward residual error is the same as the solution given the backward residual error [3]. However, averaging the two solutions is a more robust approach since we are dealing with a limited sample of the true sequence. We solve for reflection coefficients which minimize the weighted sum of the squared forward and backward residual errors averaged over the sequence, i.e.,

$$E_\mu = \sum_{t=\mu+1}^{\tau} \left[(e_{\mu,t})^T Q^f e_{\mu,t} + (b_{\mu,t})^T Q^b b_{\mu,t} \right], \quad (11)$$

where matrices Q^f and Q^b weight the impact of the forward and backward components. The relative accuracy of the lower-order forward and backward models provides confidence measures for current iteration. The higher the covariance of the driving distribution, the more uncertainty in the model and therefore the less confidence we have in the resulting estimates for the reflection coefficients. We set the weights to the inverse of the covariance of the driving distribution for the forward and backward models of order $M-1$, called the *power matrices*¹, i.e.,

$$Q^f = (P_{\mu-1}^f)^{-1}, \quad Q^b = (P_{\mu-1}^b)^{-1}, \quad (12)$$

where,

$$P_{\mu-1}^f = [R_0 \ R_1 \ \dots \ R_{\mu-1}] \mathbf{F}_{\mu-1}, \quad (13)$$

$$P_{\mu-1}^b = [R_{\mu-1} \ R_{M-2} \ \dots \ R_0] \mathbf{B}_{\mu-1}. \quad (14)$$

and R_i is the sample autocorrelation of the observed sequence under the assumption of stationarity,

$$R_i = \frac{1}{\tau - \mu} \sum_{t=\mu+1}^{\tau} x_t (x_{t-i})^T. \quad (15)$$

The power matrices are positive definite, and therefore invertible, in any physically realizable linear dynamic system [23]. Using nonsingular weight matrices provides a unique solution to the minimization of (11) [24]. Moreover, choosing such weights simplifies the equation significantly. By taking the derivative of E_μ with respect to the reflection coefficients $F_{\mu,\mu}$ and using weights $P_{\mu-1}^f$ and $P_{\mu-1}^b$, the following is derived in [24],

$$H F_{\mu,\mu} + P_{\mu-1}^b F_{\mu,\mu} (P_{\mu-1}^f)^{-1} D = -2G, \quad (16)$$

which one can use to solve for $F_{\mu,\mu}$. D and H are the covariance of the offset forward and backward residuals respectively, and G is the correlation between the offset residuals:

$$D = \sum_{t=1}^{\tau-\mu} \epsilon_{\mu,t} (\epsilon_{\mu,t})^T, \quad H = \sum_{t=1}^{\tau-\mu} \beta_{\mu,t} (\beta_{\mu,t})^T, \quad G = \sum_{t=1}^{\tau-\mu} \beta_{\mu,t} (\epsilon_{\mu,t})^T. \quad (17)$$

¹In the forward model shown in equation (4), WW^T is the power matrix.

The forward and backward offset residuals are defined as follows²,

$$\varepsilon_{\mu,t} = x_{t+\mu} + \sum_{i=1}^{\mu-1} F_{\mu-1,i} x_{t+\mu-i}, \quad (18)$$

$$\beta_{\mu,t} = x_t + \sum_{i=1}^{\mu-1} B_{\mu-1,i} x_{t+i}. \quad (19)$$

We solve for $B_{\mu,\mu}$ using the generalized conjugate relationship [3],

$$B_{\mu,\mu} = (P_{\mu-1}^f)^{-1} (F_{\mu,\mu})^T P_{\mu-1}^b. \quad (20)$$

From (9), (10), (13), (14), and (20), the following recursive updates can be derived for the power matrices [3],

$$P_M^f = P_{\mu-1}^f - (F_{\mu,\mu})^T P_{\mu-1}^b F_{\mu,\mu}, \quad (21)$$

$$P_M^b = P_{\mu-1}^b - (B_{\mu,\mu})^T P_{\mu-1}^f B_{\mu,\mu}. \quad (22)$$

Using this recursive definition, rather than (13) and (14), the higher-order autocorrelation estimates are not calculated from the sample sequence.

To initialize the algorithm, in the zero-order model we assume the sequence is the output of the stochastic component of the model. Therefore $P_0^f = P_0^b = R_0$, $\varepsilon_{0,t} = x_{t+1}$ and $\beta_{0,t} = x_t$.

To summarize MESA: Given the coefficients for a model of order $\mu - 1$, $\mathbf{F}_{\mu-1}$, and the state-space projection, X_1^τ , of the observed image sequence, (13) and (14) are used to determine the power matrices, $P_{\mu-1}^f$ and $P_{\mu-1}^b$, and (18) and (19) solve for the offset residuals, $\varepsilon_{\mu,t}$ and $\beta_{\mu,t}$. The forward reflection coefficients $F_{\mu,\mu}$, which minimize the squared sum of weighted residual errors, are determined by (16) and the backward reflection coefficients $B_{\mu,\mu}$ are calculated using the generalized conjugate relationship (20). Using the reflection coefficients $F_{\mu,\mu}$ and $B_{\mu,\mu}$, and the lower-order model parameters $\mathbf{F}_{\mu-1}$, (9) and (10) provide the coefficients \mathbf{F}_μ for a model of order μ .

4 Results

There are several ways one can evaluate and compare synthesized image sequences [1]. Here we use the one-step prediction error to quantify the quality of our results, as in [6],

$$err_\mu(i) = \left\| y_i + C \left(\sum_{j=1}^{\mu} F_{\mu,j} (C^\circ y_{i-j}) \right) \right\|_2, \quad (23)$$

where $C^\circ \equiv C^T (CC^T)^{-1}$ is the pseudo-inverse of C .

Higher-order dynamic models are shown to improve the average one-step prediction error for the test sequences in Fig. 1. As more temporal information is used to generate subsequent image frames, the prediction error of the synthesized images decreases.

²The notation for the indices of the offset residuals is somewhat counter-intuitive. Nonetheless, it is used throughout time-series literature and, for consistency, it will be used here as well. Residuals $\varepsilon_{\mu,t}$ and $\beta_{\mu,t}$ use the estimation from models of order $\mu - 1$, however a different interval of states is used within the calculation.

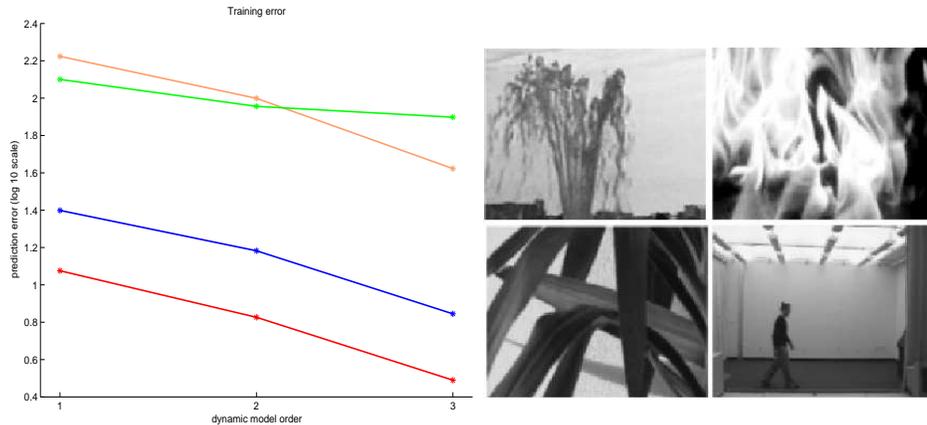


Figure 1: The effect of changing the order μ of the dynamic model is shown for four sequences: the fountain sequence [25] (blue), the fire sequence [25] (yellow), the house plant sequence [11] (green), and the walking sequence [19] (red). A frame of each sequence is shown on the right. The house plant sequence was trained with $\tau = 200$ frames and the others with $\tau = 80$. Appearance model consisted of 25-dimensions. The one-step prediction error was average over all $\tau - \mu$ sets of initialization frames.

Depending on the type of motion in the scene, the advantage of second and third-order dynamic models varies. In the house plant sequence the oscillatory swaying motion of the leaves is not captured by first-order dynamics but can be modeled using second-order dynamics. Third-order dynamics, however, do not provide much further improvement. This improvement is illustrated on the left in Fig. 2. The effect of changing the length of the sequence used for training the dynamical model, is also shown in Fig. 2. For each length the mean error was calculated from 20 models trained on different intervals of the original sequence. For each model, the median³ of the one-step prediction error is calculated over 40 initialization intervals sampled from the original sequence at regular intervals.

Although convenient for optimization, the one-step prediction error alone is not sufficient for evaluating of the overall quality of a synthesized sequence. Without the ability to consider extended intervals of time, the stability of the system is not captured. Moreover, the mean-squared error does not measure perceptual quality. For example, increasing the dimension of the appearance model decreases the prediction error, but beyond some small dimension there was no difference in perceptual quality for most textured sequences.

We found a higher-order dynamical model to be necessary to capture pendulum-like movement, such as the swaying of the leaves in the house plant sequence. In the synthetic sequences generated by a first-order model the leaves flicker, whereas, the sequences generated by a second-order model capture the swaying motion. In order to explore this, one can analyze the temporal frequency of the image intensities; we expect the jittery motion to exhibit more power at high-frequencies than the swaying motion. A set of image positions were randomly sampled according to a uniform distribution. The image sequence was spatially blurred by a Gaussian and then measured at the sampled locations.

³The median was used to accommodate short intervals of frames and increase robustness to the few instances when the power matrix was ill-conditioned causing the error to explode.

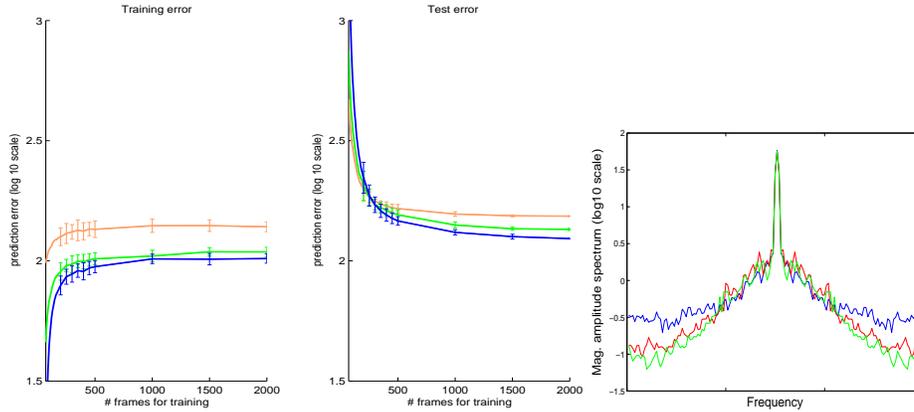


Figure 2: Results for the synthesized house plant sequence. LEFT: The effect of increasing the order of the dynamic model on house plant sequence syntheses. The one-step prediction error results reflect the visually observed results: increasing from a 1st-order (yellow) to a 2nd-order (green) dynamic model improves accuracy of the synthesized sequence more than increasing from a 2nd-order to a 3rd-order (blue) model. Models used appearance models of 25-dimensions and training lengths from 70-2000 frames. RIGHT: Average magnitude of the amplitude spectrum. The larger amount of high frequency information in the 1st-order model (blue) is in loose agreement with the perception of the jittery motion in the video. The results from the 2nd-order model (red) more closely resemble the training data (green).

After taking the Fourier transform of the resulting temporal signal, the magnitude of the frequency was averaged over all sampled positions to obtain one generalized signal for each synthesized sequence. A cosine temporal window was used before taking the Fourier transform to reduce windowing effects. The average amplitude spectrum for the first and second-order synthesized sequences of the house plant video are shown on the right in Fig. 2. The larger amount of high frequency information in the first-order model is in loose agreement with the perception of the jittery motion in the video.

When the autoregressive model is provided with a sufficient number of frames for training, relative to MESA, the Yule-Walker method finds parameters which generate images with a smaller one-step prediction error in the first few frames. A full sequence cannot be generated using these parameters, however, because the predictions become inaccurate over time due to model instability. The stability of models learned with MESA is guaranteed, but the results of the model are not necessarily perceptually accurate. In particular, without a sufficient amount of training data, the power matrices are ill-conditioned and the error is significant. It is important to note, however, that neither the Yule-Walker method nor MESA will provide a useable model under such conditions.

4.1 Linear Model Limitations

Our results demonstrate that a significant amount of the movement in the scene can be captured with a linear autoregressive model, especially with higher-order dynamics. However, real-world visual scenes exhibit complex dynamics. As expected, there are non-linear components within most observed motions which are not well described by our

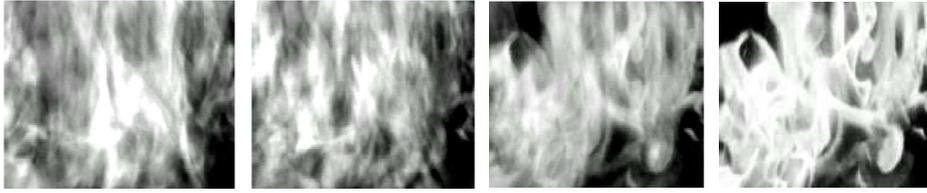


Figure 3: The higher-order dynamic models produce syntheses which resemble the original data over a longer interval. From left to right, frame 52 of the flame sequence synthesized from 1st, 2nd, 3rd-order dynamic models, and the corresponding frame from the original sequence.

model.

The deterministic component of the dynamic model provides a linear prediction of the subsequent state and the final estimation lies within a multidimensional Gaussian distribution centered at this prediction. Similar images occupy a more complex manifold in the subspace and learning the manifold may require a lot of data to ensure a dense sample of the image space [17]. Using linear dynamics with a Gaussian driving distribution will not guarantee that predicted states remain on this manifold. Moreover, because the image dataset is not convex slight inaccuracies in the prediction cause dispersion artifacts in the synthesis. For example, in the fire sequence synthesis the flame filaments are distinct and compact initially, like the original sequence. As the length of the synthesis increases, the state predictions decrease in accuracy, drift further from the manifold and the flames are dispersed across the image plane. As the order of the model increases, however, the syntheses resemble the original data over a longer interval, as shown in Fig. 3.

5 Conclusion

The results of this work illustrate how higher-order dynamics contribute to the perceptual accuracy of the novel synthesized sequences generated by autoregressive models. The complicated motion patterns which extend over multiple frames of dynamic textures are more adequately represented when additional temporal information is provided during the learning process and when generating the motion in the scene.

Without sufficient training data, previously used techniques for learning autoregressive model parameters produced unstable and inaccurate results, in particular when using higher-order dynamic models. To overcome this limitation we applied MESA, a linear prediction technique common in control theory literature which generates a reliably stable autoregressive model.

Dynamic textured sequences are complicated scenes with complex motion patterns. We have found that a significant amount of the perceptually relevant information in the scene is captured by higher-order linear autoregressive models. The models explored in this work could be used either for an accurate prediction of a few frames ahead in the sequence or to capture a general description of the motion upon which more detail could potentially be incorporated. The latter opens an interesting direction for future research.

References

- [1] See supplementary material; URL: <http://www.cs.toronto.edu/~mhyndman>
- [2] Z. Bar-Joseph, R. El-Yaniv, D. Lischinski, and M. Werman. Texture mixing and texture movie synthesis using statistical learning. *Vis. Comp. Graph.*, 7(2):120–135, 2001.
- [3] J. P. Burg. *Maximum Entropy Spectral Analysis*. PhD thesis, Stanford University, 1975.
- [4] Antoni B. Chan and Nuno Vasconcelos. Probabilistic kernels for the classification of autoregressive visual processes. In *IEEE CVPR*, 1:846–851, 2005.
- [5] J. S. De Bonet. Multiresolution sampling procedure for analysis and synthesis of texture images. *Comp. Graphics*, 31:361–368, 1997.
- [6] G. Doretto, A. Chiuso, Y. Wu, and S. Soatto. Dynamic textures. *IJCV*, 51(2):91–109, 2003.
- [7] G. Doretto and S. Soatto. Editable dynamic textures. In *ACM SIGGRAPH Sketches*, 2002.
- [8] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH*, pp. 341–346, 2001.
- [9] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. *IEEE ICCV*, 2:1033–1038, 1999.
- [10] D. J. Heeger and J. R. Bergen. Pyramid-based texture analysis/synthesis. *SIGGRAPH*, pp. 229–238, 1995.
- [11] Midori Hyndman. MSc Thesis, University of Toronto, 2006.
- [12] B. Julesz. Visual pattern discrimination. *IRE Trans. Inform. Theory*, 8(2):84–92, 1962.
- [13] M. Kaveh and G. Lippert. An optimum tapered burg algorithm for linear prediction and spectral analysis. *IEEE Trans. ASSP*, 31(2):438–444, 1983.
- [14] V. Kwatra, A. Schodl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: image and video synthesis using graph cuts. *ACM Trans. Graph.*, 22(3):277–286, July 2003.
- [15] L. Liang, C. Liu, Y. Xu, B. Guo, and H. Shum. Real-time texture synthesis by patch-based sampling. *ACM Trans. Graph.*, 20(3):127–150, July 2001.
- [16] M. Morf, B. Dickinson, T. Kailath, and A. Vieira. Efficient solution of covariance equations for linear prediction. *IEEE Trans. ASSP*, 25(5):429–433, 1977.
- [17] S. Nayar, H. Murase, and S. Nene. Parametric appearance representation. In *Early Visual Learning*, pp. 131–160. Oxford University Press, 1996.
- [18] R. C. Nelson and R. Polana. Qualitative recognition of motion using temporal texture. *CVGIP: Image Underst.*, 56(1):78–89, July 1992.
- [19] Georgia Tech. GVU Centre/College of Comp. *Human Identification at a Distance* website.
- [20] J. Portilla and E. P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *IJCV*, 40(1):49–70, 2000.
- [21] P. Saisan, G. Doretto, Y. Wu, and S. Soatto. Dynamic texture recognition. *IEEE CVPR*, 2:58–63, 2001.
- [22] A. Schodl, R. Szeliski, D. Salesin, and I. Essa. Video textures. *SIGGRAPH*, pp. 489–498, 2000.
- [23] G.A. Smith and A.J. Robinson. A comparison between the EM and subspace identification algorithms for time-invariant linear dynamical systems. Tech. Report, Cambridge Univ., 2000.
- [24] O. Strand. Multichannel complex maximum entropy (autoregressive) spectral analysis. *IEEE Trans. on Automatic Control*, 22(4):634–640, 1977.
- [25] M. Szummer and R. W. Picard. Temporal texture modeling. *ICIP*, 3:823–826, 1996.
- [26] P. Van Overschee and B. De Moor. N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1):75–93, 1994.
- [27] L. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. *SIGGRAPH*, pp. 479–488, 2000.
- [28] Y. Xu, B. Guo, and H. Shum. Chaos mosaic: Fast and memory efficient texture synthesis. Techn. Report, Microsoft Research, April 2002.