Time Series Prediction by Chaotic Modeling of Nonlinear Dynamical Systems

Arslan Basharat*+
*Kitware Inc.
Clifton Park, NY, USA
arslan.basharat@kitware.com

Mubarak Shah+
+University of Central Florida
Orlando, FL, USA
shah@cs.ucf.edu

Abstract

We use concepts from chaos theory in order to model nonlinear dynamical systems that exhibit deterministic behavior. Observed time series from such a system can be embedded into a higher dimensional phase space without the knowledge of an exact model of the underlying dynamics. Such an embedding warps the observed data to a strange attractor, in the phase space, which provides precise information about the dynamics involved. We extract this information from the strange attractor and utilize it to predict future observations. Given an initial condition, the predictions in the phase space are computed through kernel regression. This approach has the advantage of modeling dynamics without making any assumptions about the exact form (linear, polynomial, radial basis, etc.) of the mapping function. The predicted points are then warped back to the observed time series. We demonstrate the utility of these predictions for human action synthesis, and dynamic texture synthesis. Our main contributions are: multivariate phase space reconstruction for human actions and dynamic textures, a deterministic approach to model dynamics in contrast to the popular noise-driven approaches for dynamic textures, and video synthesis from kernel regression in the phase space. Experimental results provide qualitative and quantitative analysis of our approach on standard data sets.

1. Introduction

We propose a new approach to model and predict time series data observed in different types of videos. Such data would comprise of a sequence of observations over time, for instance, joint location or angle of a particular human body joint, pixel intensity at a particular location, etc. These time series would typically be generated by a deterministic nonlinear dynamical system with known initial condition. A good model of the underlying dynamics is important for predictions that are used in applications like video synthesis. When synthesizing longer sequences from a short sample video, it is desirable to generate realistic and smooth transitions. A trivial approach would be to concatenate the sample video multiple times, but this results in non-realistic transitions. Fig. 1 shows an example of a scalar time series signal from running action. This data is from one of the three dimensions corresponding to the 3D location of the human foot. The predicted signal (broken red) generated by the proposed approach creates a smooth transition and continues to depict the same dynamics as earlier. Such a mechanism could be useful in synthesizing repetitive human actions and dynamic textures for long durations. This can have a variety of applications in computer vision and graphics including: human motion animation, noise handling from motion capture data, more realistic dynamic texture synthesis, etc.

This paper presents a novel approach for synthesizing such sequences using the relevant concepts from dynamical systems and chaos theory. In dynamical systems the time evolution of data points is defined in some higher dimensional *phase (or state) space*. Chaos theory is related to the study of *chaotic systems*; that is, nonlinear dynamical systems that exhibit deterministic behavior with a known *initial condition* (starting point). Human actions such as walking, running, jumping, etc. have been studied before by Ali et al. [1] and are found to exhibit the deterministic properties of the chaotic systems. The observed scalar time series signals are transformed into a higher dimensional phase space through delay reconstruction (see Sec. 2.1). This results in a *strange attractor* which is characteristic of the underlying chaotic system. Note that a chaotic signal can be irregular and less predictable in the observed time series space, while in phase space it has a regular structure due to its deterministic nature. For prediction in phase space, several regression techniques can be used to compute the temporal mapping function. Many of these techniques often assume a particular underlying form of the mapping function (linear, polynomial, radial basis function etc.). However, in case of human actions and dynamic textures we are not aware of the exact forms of the mapping functions responsible for generating the dynamics. Hence, instead of approximat-

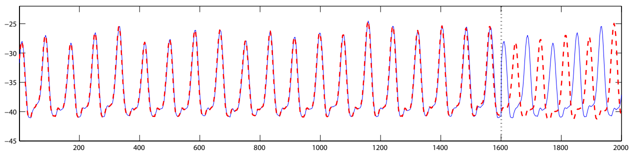

Figure 1. Abrupt vs. smooth transition: Original time series signal (solid blue) is repeated at the 1600 mark where it shows an abrupt transition. The predicted signal (broken red) shows a smooth transition and synthesizes the signal persistently.

ing a the functional form from the observed data, we rely on a more general approach. We use a nonparametric data driven model, based on kernel regression [16], to predict the future points along the strange attractor. These predictions are then transformed back into time series of longer duration with continuous motion. In order to generate more realistic and synchronized multiple time series signals, we investigate the use of multivariate vs. univariate reconstruction for prediction. The use of multivariate time series embedding for human actions and dynamic textures is novel. The predicted time series signals of body-pose parameters are used to synthesize and track human motion. In addition, the predicted pixel intensities are used to synthesize dynamic texture sequences.

The aim of this paper is to investigate the relevant concepts from chaos theory and propose a novel and robust model for video synthesis. The novelty of this work lies in:

- The formulation of phase space reconstruction from the multivariate time series data of human actions and dynamic textures. Previously [1], only univariate phase space models of human actions have been studied for action recognition.
- A new deterministic dynamical model for dynamic textures in contrast to previously popular stochastic noise-driven dynamical systems [9, 24].
- A new nonparametric model based on kernel regression in phase space.

We also provide experimental validation of viability of chaotic modeling approach for video synthesis. We show that our approach outperforms many recent approaches for dynamic textures synthesis.

1.1. Related Work

Polana and Nelson [17] classified visual motion into three classes: motion events, activities, and temporal textures. Motion events (e.g. sitting, opening window) don't exhibit temporal or spatial periodicity. Activities (e.g. walking, jumping) are formed by the motion patterns that are periodic in time and localized in space. Temporal textures (e.g. waves on water surface, smoke) present statistical regularity but have indeterminate spatial and temporal extent. In this paper we focus on the temporal regularity of the last two classes. For this we rely on the powerful tools from chaos theory to model deterministic dynamical systems [13].

In computer vision, dynamical systems have been used in a variety of applications, including human motion (action) modeling [1, 2, 3, 10] and dynamic textures [7, 9, 12, 15, 24, 23, 20].Most of these approaches model underlying system dynamics by using linear systems, while others use nonlinear dynamical systems. In many cases, nonlinear approaches provide a more accurate model but have to approximate the parametric form of the underlying system. This parameter learning may be imprecise and that can be a source of error. Our approach belongs to the category of the nonlinear dynamical systems that use nonparametric model, which therefore do not require parameter learning.

Human actions have been modelled by a nonparametric chaotic system by Ali et al. [1]. They proposed the nonparametric chaotic model for human actions and demonstrated the viability for action recognition. We extend their univariate delay embedding model of human action to the multivariate case. This model is then used for predictions that are used for synthesis. Wang et al. [10] have presented another strong model for human motion. They propose a nonparametric dynamical system based on Gaussian processes. This approach is only demonstrated for human motion and not for the higher dimensional data, such as dynamic textures. The case of dynamic textures is more challenging than human action because of the higher dimensional observations and more irregular variations in the system state. Our approach is general enough to be applicable to both human actions and dynamic textures. In addition, our method does not require multiple exemplars for training in order to learn a particular action, making it more practical.

Many of the previous approaches for dynamic texture rely on stochastic noise-driven linear [9, 24] and nonlinear [7] dynamical systems. Instead, we show that the typical dynamic textures can be modelled accurately by deterministic dynamical systems. The detailed experimental validation proves our argument. In [14] and [15], authors present approaches for learning nonlinear manifold for the observed time series. We have compared our method with [15] and show that our approach generates more realistic dynamic textures, because it does not suffer from the errors due to imprecise learning.

Time series modeling and prediction has been an active area of research due to the wide variety of applications in the financial market, weather, biology, etc. The initial approaches typically relied on AR, MA, or ARMA univariate models. More sophisticated approaches rely on nonlinear modeling [6] and state space projection of the time series [18]. Our approach has both of these properties. Ralaivola et al. [18] present an approach for time series prediction based on kernel trick and support vector regression. In comparison, our approach is based on delay embedding [22]

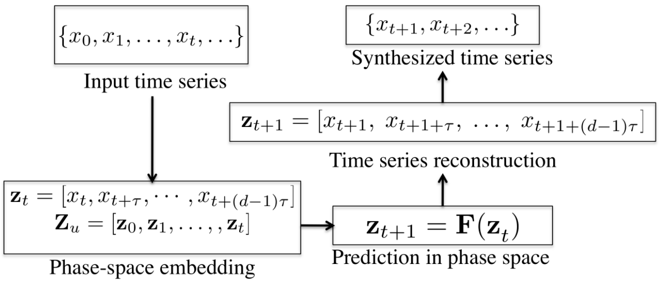

Figure 2. Main steps of the proposed approach for time series synthesis.

and kernel regression [16]. Delay embedding generates the unique *strange attractor* that can be used for system modeling and classification. [13].

2. Proposed Approach

We investigate dynamical systems that define the time evolution of underlying dynamics in a phase (or state) space. First task is to find a way for phase space reconstruction from times series. The time series observations $\{x_0, x_1, \ldots, x_t, \ldots\}$ are transformed to the phase space vectors $\{\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_t, \ldots\}$ through delay embedding, which is explained in Sec. 2.1. In the case of deterministic nonlinear dynamical (chaotic) systems, specifying a point in the phase space identifies the state of the system and vice versa. This implies that we can model the dynamics of a system by modeling the dynamics of the corresponding points in the phase space [13]. This idea forms the foundation of modeling the underlying chaotic system of unknown form and predicting future states. A system state is defined by a vector $\mathbf{z}_t \in \mathbb{R}^n$. The dynamics of these states are defined either by an n-dimensional mapping function

$$\mathbf{z}_{t+1} = \mathbf{F}(\mathbf{z}_t), \tag{1}$$

or by n first order differential equations. The latter approach is typically used for studying theoretical systems because the exact equations are rarely known for the experimental systems. The former approach, which is based on the mapping function, is more popular for the experimental systems. Sec. 2.2 describes a kernel regression based mapping function that we adopt for predicting future system states. These new states are transformed back to output time series as explained in Sec. 2.3.

2.1. Phase Space Reconstruction

Phase space reconstruction is performed by the delay embedding of the observed data into phase space vectors. The details of the univariate delay embedding for human actions are provided by Ali et al. [1], however, we include relevant information for completion. Takens' delay embedding theorem forms the basis of this approach [22]. It states that *a map exists between the original state space and a reconstructed state space*. The theorem shows that the dynamical properties of the system from the true state space are preserved through the embedding transformation. Therefore, the delay vectors $\mathbf{z}_t = [x_t,\ x_{t+\tau},\ \ldots,\ x_{t+(d-1)\tau}] \in \mathbb{R}^d$, generate the phase space. The two parameters to be computed are lag τ and embedding dimension d.

The most popular approach for computing lag τ is based on the amount of mutual information between x_i and $x_{i+\tau}$ pair of observed values. The basic idea here is to look for the minimum τ for which the mutual information between observations is lowest. The details of the algorithm are available in [11]. Fig. 3(a) shows a univariate time series from one of the three dimensions of the foot of a running person. Fig. 3(b) shows the plot of possible τ values vs. amount of mutual information. The point of the first local minima of this plot is chosen as the lag τ. The optimal embedding dimension d can be computed by using the false nearest neighbors method proposed in [4]. The basic idea of this method is to find the smallest d, while minimizing the number of false nearest neighbors due to dimension reduction. Fig. 3(c) shows the plot of possible values of d vs. fraction [0,1] of the points that do not have false nearest neighbors. Note that the fraction converges to 1 (100%) at $d = 5$, so choosing $d > 5$ would not be an optimal choice.

The values of τ and d are used to transform the univariate time series into the phase space (or delay) vectors $\mathbf{z}_t$ stacked as

$$\mathbf{Z}_u = \begin{pmatrix} \mathbf{z}_0 \\ \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} x_0 & x_\tau & \cdots & x_{(d-1)\tau} \\ x_1 & x_{1+\tau} & \cdots & x_{1+(d-1)\tau} \\ x_2 & x_{2+\tau} & \cdots & x_{2+(d-1)\tau} \\ & & \vdots & \end{pmatrix}. \tag{2}$$

Note that each observed scalar value is repeated several time in this matrix. The sequence of the rows in this embedding matrix is important as it generates a trajectory in the phase space. Fig. 4(a) shows the 3D projection of 5D phase space for the time series presented in Fig. 3. This blue trajectory forms the *strange attractor* in the phase space. The metric, dynamical, and topological properties of this strange attractor are characteristic of the underlying nonlinear dynamical

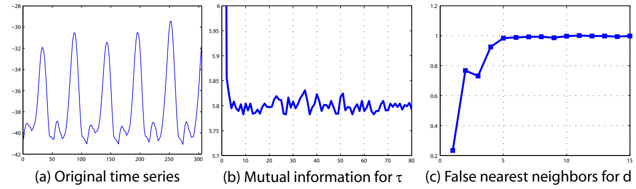

Figure 3. Steps for phase space reconstruction. (a) The observed univariate time series. (b) Mutual information plot to determine minimum delay (first local minimum, $\tau = 9$). (c) The embedding dimension is computed by finding the smallest value that gives a small number of false nearest neighbors (converging to 1, $d = 5$).

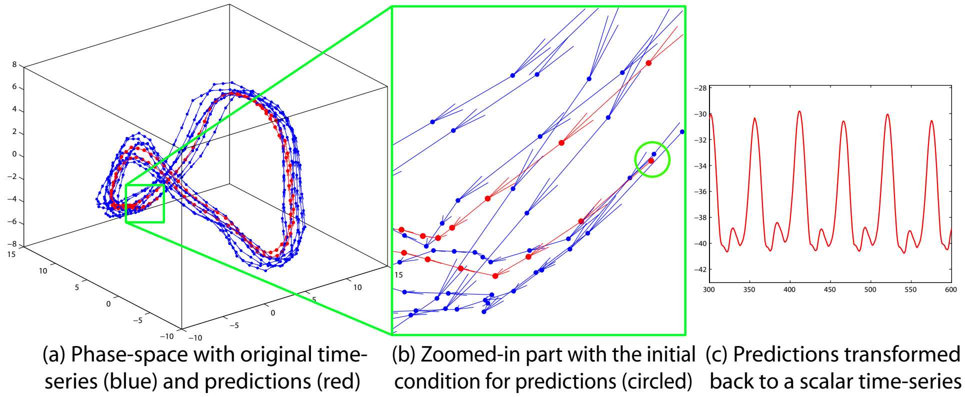

Figure 4. Predicting dynamics of a time series. Original time series is transformed into a strange attractor in the phase space. Kernel regression is used to estimate predicted values following behavior of neighbors. The predicted points in the phase space are transformed into a synthesized time series.

system [13]. We will be relying on modeling the evolution (flow) of the observed points along this strange attractor to predict the future locations.

This form of the embedding $\mathbf{Z}_u$ is feasible for prediction in the case of univariate time series. However, in computer vision we frequently observe time series generated by a dynamical system that involves multiple variables (dimensions) simultaneously. For instance, during human motion directly connected body joints impose certain constraints on the motion of each other. Similarly, in the case of dynamic textures the pixels values in the same neighborhood evolve together. The trivial solution would be to proceed with performing univariate prediction separately for each dimension of the time series. We demonstrate through experiments that this approach breaks down due to the dependence between joint locations and neighboring pixels. Hence, a phase space reconstruction is desirable where prediction is performed for all the dimensions of a multivariate time series simultaneously. Cao et al. [5] have shown that a simple yet powerful extension of the univariate embedding can be useful for the multivariate time series prediction. For a multivariate time series, with observations $\mathbf{x}_t = [x_{1,t},\ x_{2,t},\ \ldots,\ x_{D,t}]^T \in \mathbb{R}^D$, an appropriate phase space $\mathbf{Z}_m = [\mathbf{z}_0, \mathbf{z}_1, \mathbf{z}_2, \ldots]^T$ would be created by a set of delay vectors redefined as

$$\begin{aligned}\mathbf{z}_t = [&x_{1,t},\ x_{1,t+\tau_1},\ \ldots,\ x_{1,t+(d_1-1)\tau_1},\\ &x_{2,t},\ x_{2,t+\tau_1},\ \ldots,\ x_{2,t+(d_2-1)\tau_2},\\ &\ldots,\\ &x_{D,t},\ x_{D,t+\tau_1},\ \ldots,\ x_{D,t+(d_D-1)\tau_D}] \in \mathbb{R}^{\sum_{i=1}^{D} d_i}.\end{aligned} \quad (3)$$

Here τ_i and d_i are respectively the delay and the embedding dimension for each one of the D dimension of time series. $\mathbf{z}_t$ maps to a point in the higher dimensional phase space and is linked to the next point $\mathbf{z}_{t+1}$ by the order in $\mathbf{Z}_m$ matrix. Fig. 4(b) shows such points highlighted by dots and connected through arrows showing the direction of evolution.

2.2. Prediction in Phase Space

In order to perform prediction we need to compute the mapping function $\mathbf{F}$ (Eqn. 1). The exact form of $\mathbf{F}$ is unknown in case of general human motions or dynamic textures. The "appropriate" selection of the model poses a challenge when one is not aware of the exact physics of the underlying dynamics. One popular form of the model is given by

$$\mathbf{z}_{t+1} = \mathbf{F}(\mathbf{z}_t) = \sum_{m=1}^{M} \mathbf{c}(m,t)\phi_m(\mathbf{z}_t), \quad (4)$$

which is a linear combination of M possibly nonlinear functions ϕ_m with $\mathbf{c}(m,t)$ providing the coefficients. ϕ_m are usually chosen to be polynomials, radial basis functions, or logarithmic functions while the coefficient values $\mathbf{c}(m,t)$ are computed during functional approximation (e.g. least squares).

We avoid guessing a particular model by using a non-parametric model based on kernel regression [16]. The main idea is to estimate the mapping function using a weighted average of dynamics of neighboring points in the phase space. Hence, the mapping is given by

$$\mathbf{z}_{t+1} = \mathbf{F}(\mathbf{z}_t) = \sum_{k=1}^{N_n(\mathbf{z}_t)} (\mathbf{y}_{k+1} - \mathbf{y}_k + \mathbf{z}_t) w_k(\mathbf{z}_t, \mathbf{y}_k), \quad (5)$$

where $\mathbf{y}_k$ is one of the $N_n(\mathbf{z}_t)$ nearest neighbors of $\mathbf{z}_t$. Each of these neighbors has a corresponding next point $\mathbf{y}_{k+1}$ in the phase space. As shown in Fig. 4(b), the vectors between the consecutive points are used in the neighborhood. The weights are computed from the kernel which is a decreasing function of distance from the reference point. Nadaraya-Watson [16] defined these weights as

$$w_k(\mathbf{z}_t, \mathbf{y}_k) = \frac{K_h(||\mathbf{z}_t - \mathbf{y}_k||)}{\sum_{k=1}^{N_n(\mathbf{z}_t)} K_h(||\mathbf{z}_t - \mathbf{y}_k||)},\ K_h(b) = \frac{1}{h}K\left(\frac{b}{h}\right), \quad (6)$$

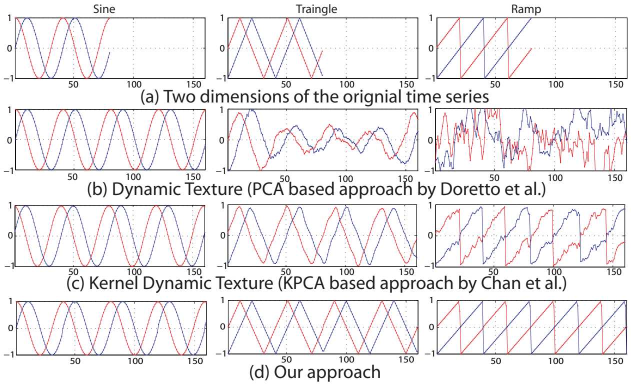

Figure 5. Comparison on synthetic data. (a) Sine, triangle, and ramp input time series. (b) and (c) show the synthesized output by Doretto et al.'s Dynamic Textures [9] and Chan et al.'s Kernel Dynamic Textures [7] respectively. (d) Synthesized output of our method provides more accurate reconstruction for all three signals.

where K is the kernel function which can be Guassian, Epanechnikov, etc, h is the bandwidth of the kernel and can be used for over smoothing. In our experiments we use $\mathcal{N}(0,1)$ kernel and bandwidth $h = 0.5$. Such a chaotic modeling approach is generally: quite robust to noisy data, more accurate in experimental systems, and good for prediction while preserving important invariants of the dynamics [13]. Such an approach has the advantage of capturing a desirable balance between local and global parametric regression approaches. Local models are known to have the problem of large computational and memory requirements. On the other hand, the global models over generalize while computing one functional representation that models the whole attractor in the phase space.

Fig. 4 shows the phase space reconstruction and predictions from the time series shown in Fig. 3(a). The predictions are shown by red trajectories along with their directions of flow. Fig. 4(b) shows the starting point (initial condition) of the prediction with closest neighboring points that contribute the most (through symmetric kernel) to the first prediction. Note that the first resultant arrow follows the immediate neighbors very closely. The predicted trajectory keeps evolving along the strange attractor following the system dynamics.

2.3. Time Series Reconstruction

To recover a time series from the predictions in the phase space we have to extract the time series from univariate $\mathbf{Z}_u$ or multivariate $\mathbf{Z}_m$ matrices. For the univariate case $\mathbf{Z}_u$ (see Eq. 2) it is simply extracting the first column followed by last τ rows from the rest of the columns. For a Txd matrix $\mathbf{Z}_u$ this generates $T + (d-1)\tau$ time series observations

$$x_i \in \{\mathbf{Z}_u(1,i),\ \mathbf{Z}_u(k, T-j)\},$$

where $0 \le i < T, \tau \ge j > 0, 1 \le k < d$. In the multivariate case, $\mathbf{Z}_m$ matrix (see Eq. 3) contains a row of

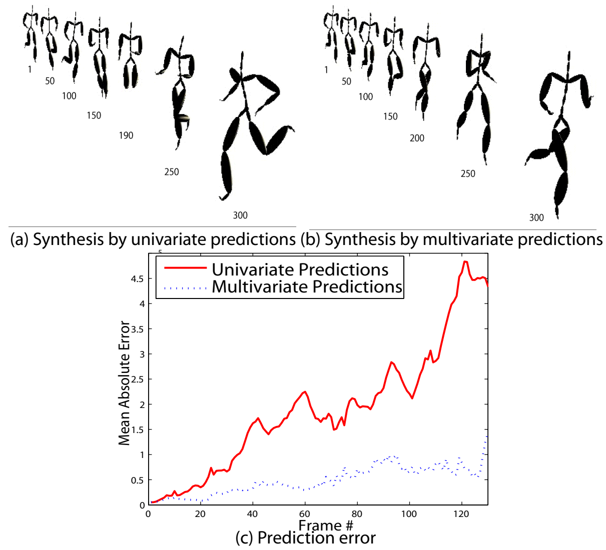

Figure 6. Univariate vs. multivariate predictions for human motion. Univariate approach (a) shows irregular poses and its global transformations while multivariate approach (b) generates a smooth sequence with all valid poses. (c) Univariate predictions also result in a higher error than the multivariate predictions.

D individual $\mathbf{Z}_u$ matrices. The multivariate time series is constructed by extracting D univariate time series from the corresponding $\mathbf{Z}_u$ as described above. Fig. 4(c) shows an example of a univariate time series extracted from the predictions in the phase space shown in Fig. 4(a). Fig. 5 shows the output of time series synthesis on three synthetic signals where $D = 2$. The embedding parameters (τ, d) are calculated to be (4, 5), (3, 4) and (5, 7) for each dimension in sine, triangle and ramp signals respectively. It shows that the output of our approach is very similar to the source signal and is better than the two recent approaches used for dynamic texture modeling [9, 7].

3. Experimental Results

The proposed approach for predicting time series is applied to human action and dynamic texture synthesis. Several experiments were performed to evaluate the performance of our approach and to compare the output with that of some of the well known methods.

3.1. Action Synthesis

We use motion capture data to acquire source time series representing the position of the body landmarks during the action. We use the motion capture data from FutureLight [1] and CMU [8] data sets for the human action synthesis. Every frame in CMU and FutureLight sequences provides a 62 and 39-dimensional body-pose descriptors respectively. CMU's descriptor is composed of bone length and joint angles, while FutureLight is composed of the absolute 3D locations of the 13 body joints. A part of the sample sequence of the human action is used to generate the observed time

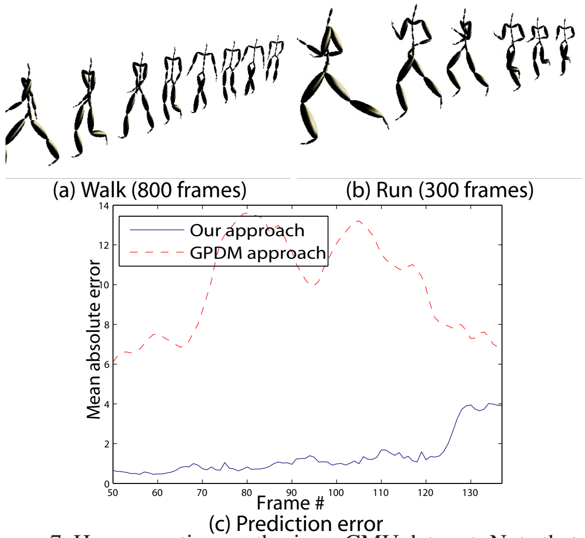

Figure 7. Human motion synthesis on CMU data set. Note that the difference between the walking and running body-poses is maintained after synthesis. (a) Every 100th frames is shown , (b) Every 50th frame is shown. (c) Quality of our predictions are compared against the ones generated by the GPDM based approach [10]. The ground truth between frame 50 and 137 is used to compute prediction error.

series $\mathbf{x}_t \in \mathbb{R}^P$, where P is the dimensionality of the body-pose descriptor. The multivariate phase space reconstruction produces $\mathbf{Z}_m$ embedding matrix for the sample action. For a given starting point $\mathbf{x}_t$, the predictions and time series reconstruction is performed as explained before. This creates a sequence $\{\mathbf{x}_t, \mathbf{x}_{t+1}, \ldots\}$ of body-pose descriptors used for final video synthesis.

We have experimented with both univariate and multivariate predictions for this task. In the univariate case, each dimension of the pose descriptor is used independently to determine the phase space reconstruction followed by prediction. In the second case, multivariate prediction approach is used to evolve the predictions in an even higher dimensional phase space (order of P-dimensional). This provides the combined evolution of different dimensions of the pose descriptor. Fig. 6 shows the keyframes from the same running sequence synthesized using the univariate (see Fig. 6(a)) and the multivariate (see Fig. 6(b)) predictions. These 300 frame long sequences have been synthesized from a 130 frames long model sequence. The keyframes in the multivariate case show normal body poses, however in the univariate case, strange poses are synthesized. Towards the end there is an unrealistic global rotation of the whole body. Fig. 6(c) shows a graph of mean absolute error in the first 130 frames from both sequences that overlap with the model sequence. This clearly shows that the proposed multivariate formulation is critical for human action synthesis.

Using the CMU data set, we show results on walking and running actions as shown in the Fig. 7. The model sequences used in our experiments are typically 100 to 500 frames long. We synthesize sequences with up to three times the original length. The highest individual embedding dimension d_i observed during experiments was 7. We also compare the accuracy of predictions with the output of GPDM based approach [10]. Fig. 7 (c) shows a graph of mean absolute error in predictions by our approach (solid blue) and by Wang et al. [10]. The sequence (CMU id : 09_04) shown in Fig. 7 (b) is used for this experiment, where frame 1-100 are used for creating the model and frame 50-137 are used to compute the error in predictions.

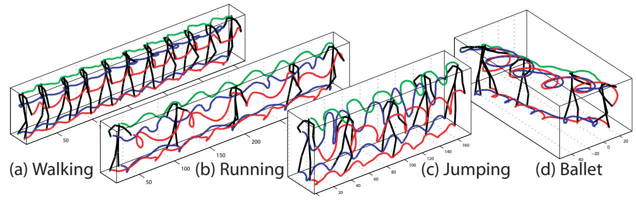

Figure 8. FutureLight data set. Synthesized sequences from each of the four different types of actions is shown. Here right hand & foot have red trajectories, left foot & hand have blue trajectories, while head has green trajectory. Faster speed in the running sequence (as compared to walking) can be noticed by the sparse stick figures that are drawn every 40 frames.

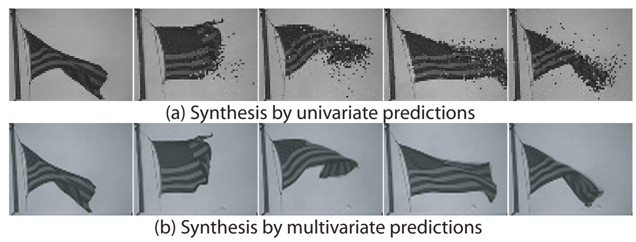

Figure 9. Dynamic texture synthesis from Stripes video. (a) Predictions of many pixels quickly become unsynchronized from the neighbors causing the noisy pixels. (b) Multivariate predictions create more realistic and smoother videos.

Using the FutureLight data set, we synthesize walking, running, jumping, and ballet actions, as shown in Fig. 8. We compute the relative locations of all other landmarks with respect to the belly (reference) point. This provides us with a 39-dimensional time series signal that will be predicted. The phase space embedding and predictions are computed through the aforementioned approach. During our experiments, the individual embedding dimension d_i would typically fall between 3 and 6 for these actions. The length of a typical model sequences used is between 220 and 500.

3.2. Dynamic Texture Synthesis

We also demonstrate the synthesis of dynamic textures through the proposed approach of chaotic modeling. The dynamic textures have stochastic regularity in the spatial and temporal extent [17]. We investigate the determinism in the structure of dynamic textures through the proposed approach. The sequence of intensity values at each pixel is

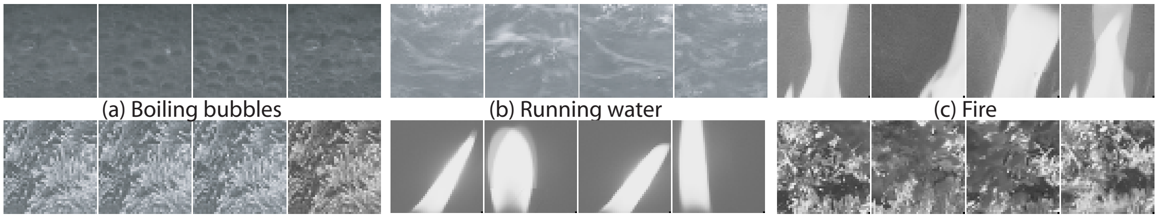

Figure 10. Dynamic texture synthesis from UCLA data set. 75 frame long model videos are used to generate 225 synthesized frames.

treated as a univariate time series, which is generated possibly by a chaotic system. We investigate the feasibility of both univariate and multivariate predictions in this case as well. The multivariate case is applied in small neighborhoods of 25x25 which creates 625-dimensional multivariate time series for each neighborhood. The actual dimensionality of the phase space would then be a sum of the individual 625 embedding dimensions d_i's. Fig. 9(a) shows the synthesized video in the case of univariate predictions. Noisy pixels become more obvious as the video progresses because predictions diverge farther from ground truth. The multivariate case Fig. 9(b) applies better spatial constraint and results in a synthesized video of better quality.

We first present synthesis results using the UCLA data set [19]. It contains 50 classes of different types of dynamic textures, including flames, trees, fountains, water etc. Each video contains 75 frames of a cropped 48x48 textured area. Each pixel provides a scalar time series, whose embedding parameters are computed individually. This is followed by multivariate phase space reconstruction and prediction. The individual embedding dimension d_i for a pixel has been observed to lie between 4 and 9 for typical dynamics of the textures used here. Fig. 10 shows a few of the synthesized frames from various types of videos in this data set.

A series of experiments have been performed to compare our approach to some of the popular approaches for dynamic texture synthesis. These include approaches by Chan et al. [7], Liu et al. [15], and Yuan et al. [24]. All of them provide means for quantitative and qualitative comparison with their approach, as well as the baseline PCA based linear dynamical system approaches and an improved version by Doretto et al. [9]. We performed experiments on the MIT dynamic textures data set [21], in order to present qualitative and quantitative comparison with these approaches. This data set contains videos that are typically 114x170 with 120 frames. These model videos were used to produce synthesized videos three times their length. The time series with pixel intensities is embedded into a higher dimensional phase space where prediction is performed. Fig. 11 presents the output of our method, along with the corresponding output of the two approaches presented in [15]. The first is a baseline approach they used which relies on simple PCA with AR model. The second is their approach based on probabilistic PCA (PPCA). In Fig. 11 we also highlight interesting area of the image with the red box. Note that both approaches in first two rows generate a ghost-like effect due to imperfect projection onto a few components, however, our approach preserves the quality. Table 1 presents quantitative comparison through mean squared error. This error is computed by the mean squared difference between the pixel values of the original and the predicted frames. We analyze the three videos (stripes, flags, and river) used in [15] and determine that our approach indeed outperforms both of these methods.

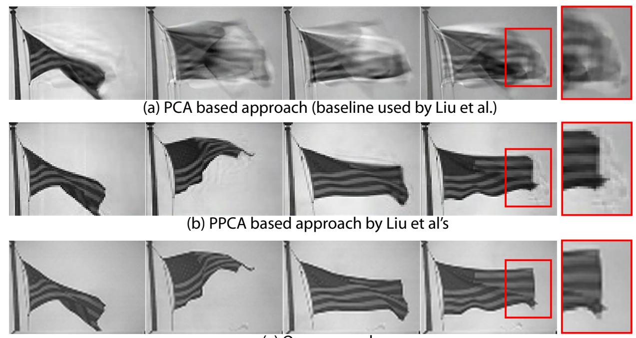

Figure 11. Dynamic texture synthesis from the Stripes video. We compare our method with the approach by Liu et al. [15] and the baseline method they used. Results obtained from our method are crisp and do not exhibit ghost-like effects, as highlighted by the red box in the last column.

Table 1. Mean squared error between the original and synthesized frames

Sequence name	Stripes (Fig. 11)	Flags	River
PCA based approach (baseline in [15])	1119.8	1445.2	1198.0
PPCA based approach [15]	2117.9	579.5	551.4
Our approach	**12.2**	**17.8**	**8.6**

Similarly, we perform another comparison with the closed-loop LDS by Yuan et al. [24], their baseline version LDS, and improved LDS by Doretto et al. [9]. Due to limited space, we only include the Fire sequence, which is the more challenging than the other two. The difference between the outputs of our approach and that from the first two approaches (basic and improved LDS) is obvious when looking at the figure. Table 2 clearly shows that our results

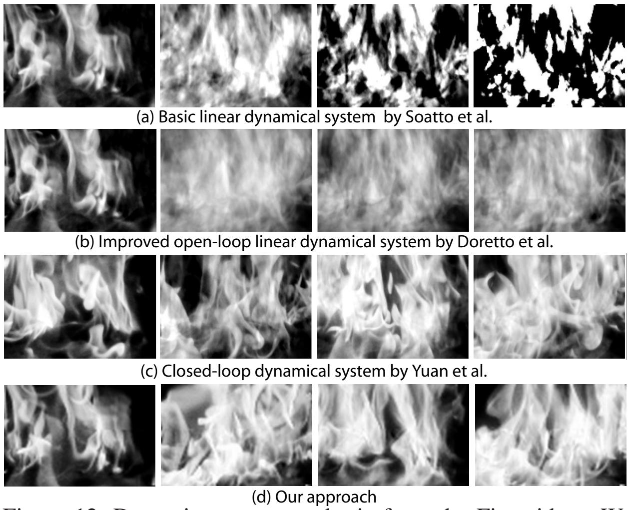

Figure 12. Dynamic texture synthesis from the Fire video. We compare our method with Yuan et al.'s [24] and the baseline they used by Doretto et al. [9].

Table 2. Mean squared error between the original and synthesized frames

Sequence name	Fire (Fig. 12)	Smoke-far	Smoke-near
Basic LDS (baseline in [24])	55264	230.7	402.6
Improved LDS [9]	55421	250.0	428.2
Closed-loop LDS [24]	1170	21.4	34.4
Our approach	**109**	**16.1**	**1.9**

are closer to the original video as compared to the out put of Yuan et al.

4. Conclusions

We have presented a new model for nonlinear dynamical systems of human actions and dynamic textures. We observed that multivariate phase space reconstruction is more suitable for predicting time series. The benefit of the multivariate reconstruction is more obvious in case of dynamic textures where the pixels are evolved together in the neighborhood. The dimension reduction approaches relying on principle components have been noticed to generate ghost-like artifacts. They can be attributed to the linear/nonlinear combination of the estimated components used for reprojection. We also show that the dynamic textures and human actions can be modelled very well by a deterministic model that is inherently different from many noise-driven models. Generalization is also another important property of our system as it is not very sensitive to the type of periodicity in time series and the parameter values. The viability, robustness, and generalization of this model has been demonstrated empirically.

Acknowledgements

This research was funded in part by the US government VACE program.

References

[1] S. Ali, A. Basharat, and M. Shah. Chaotic invariants for human action recognition. *ICCV*, 2007.

[2] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto. Recognition of human gaits. *CVPR*, 2001.

[3] C. Bregler. Learning and recognizing human dynamics in video sequences. *CVPR*, 1997.

[4] L. Cao. Practical method for determining the minimum embedding dimension of a scalar time series. *Physica D: Nonlinear Phenomena*, 1997.

[5] L. Cao, A. Mees, and K. Judd. Dynamics from multivariate time series. *Physica D: Nonlinear Phenomena*, 1998.

[6] M. Casdagli. Nonlinear prediction of chaotic time series. *Physica D: Nonlinear Phenomena*, 1989.

[7] A. B. Chan and N. Vasconcelos. Classifying video with kernel dynamic textures. *CVPR*, 2007.

[8] CMU. Dataset: `http://mocap.cs.cmu.edu/`.

[9] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto. Dynamic textures. *IJCV*, 2003.

[10] J. W. D. Fleet and A. Hertzmann. Gaussian process dynamical models for human motion. *PAMI*, 2008.

[11] A. M. Fraser. Independent coordinates for strange attractors from mutual information. *Phys. Rev.*, 1986.

[12] B. Ghanem and N. Ahuja. Phase based modelling of dynamic textures. *ICCV*, 2007.

[13] H. Kantz and T. Schreiber. Nonlinear time series analysis. *Cambridge U. Press*, 2004.

[14] R. Lin, C. Liu, M. Yang, N. Ahuja, and S. Levinson. Learning nonlinear manifolds from time series. *ECCV*, 06.

[15] C.-B. Liu, R.-S. Lin, N. Ahuja, and M.-H. Yang. Dynamic textures synthesis as nonlinear manifold learning and traversing. *BMVC*, 2006.

[16] E. A. Nadarya. On estimating regression. *Theory Pb. Appl.*, 1964.

[17] R. Polana and R. Nelson. Temporal texture and activity recognition. *Motion-Based Recognition*, 1997.

[18] L. Ralaivola and F. dAlcheBuc. Dynamical modeling with kernels for nonlinear time series prediction. *NIPS*, 2003.

[19] P. Saisan, G. Doretto, Y. Wu, and S. Soatto. Dynamic texture recognition. *CVPR*, 2001.

[20] A. Schdl, R. Szeliski, D. Salesin, and I. Essa. Video textures. *SIGGRAPH*, 2000.

[21] M. Szummer and R. W. Picard. Temporal texture modeling. *ICIP*, 1996.

[22] F. Takens. Detecting strange attractors in turbulence. *L. N. in Math*, 1981.

[23] Y. Z. Wang and S. C. Zhu. A generative method for textured motion: Analysis and synthesis. In *ECCV*, 2002.

[24] L. Yuan, F. Wen, C. Liu, and H. Y. Shum. Synthesizing dynamic texture with closed-loop linear dynamic system. *ECCV*, 2004.