Motion Texture

A Two-level Statistical Model for Motion Synthesis

Yan Li   Tianshu Wang   Heung-Yeung Shum

Microsoft Research, Asia

Motion Capture Data

Motion capture data re-use

- Motion editing
  - Interactive motion editing
    - [Bruderlin'95, Witkin'95, Lee'99, Pullen'02]
  - Adaptation to new characters and environments
    - [Hodgins'95, Gleicher'98, Popovic'99]
- Motion synthesis
  - Retain the realism of original captured data
  - Allow the user to control and direct the character

Motion synthesis

- Reordering the original data
  - Chop into motion clips & model their transitions
    [Kovar'02, Lee'02, Arikan'02], [Schodl'00] Video Texture
  - Problems
    - No generalization ability
    - Difficult to edit at the frame level
- Generative models
  - Hidden Markov Model (HMM)
    [Brand'00, Tanco'00]
  - Auto-regressive model for simple movements
    [Pavlovic'00] SLDS, [Soatto'01] Dynamic Texture

Motion texture

Representation

- A two-level statistical model
- Motion texton --- Linear dynamic system (LDS)
- Texton distribution --- Markov process

Analogous to 2D texture image

- Textons (Julesz'81)
- 2D spatial distribution of textons

Motion texture analysis/synthesis

Challenges

- How to learn the motion texture
- How to synthesize from motion texture
- How to deal with high dimensional motion data

Motion data representation

- Configuration of articulated characters
  - Linear components : $\mathbf{p}(t) \in \mathbf{R}^3$
  - Angular components : $\mathbf{q}^i(t) \in \mathbf{S}^3$

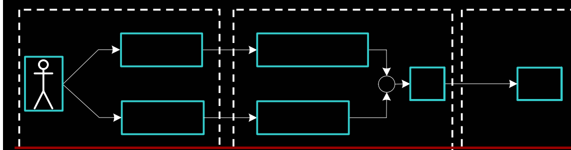
$$\mathbf{m}(t) = \begin{pmatrix} \mathbf{p}(t) \\ \mathbf{q}^1(t) \\ \mathbf{q}^2(t) \\ \vdots \\ \mathbf{q}^n(t) \end{pmatrix}$$

$\mathbf{p}(t) \rightarrow$ Position of the root joint

$\mathbf{q}^1(t) \rightarrow$ Orientation of the root joint

$\mathbf{q}^2(t), \ldots, \mathbf{q}^n(t) \rightarrow$ Orientations of the body joint

Dealing with motion data

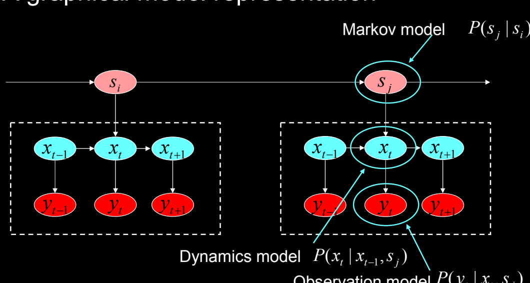


$$\mathbf{m}(t) = \begin{pmatrix} \mathbf{p}(t) \\ \mathbf{q}^1(t) \\ \mathbf{q}^2(t) \\ \vdots \\ \mathbf{q}^n(t) \end{pmatrix} \qquad \mathbf{Y}(t) = \begin{pmatrix} \mathbf{p}(t) - \mathbf{p}(t-1) \\ \log(\mathbf{q}^1(t)) \\ \log(\mathbf{q}^2(t)) \\ \vdots \\ \log(\mathbf{q}^n(t)) \end{pmatrix} \qquad \mathbf{X}(t) = \begin{pmatrix} \mathbf{x}^1(t) \\ \mathbf{x}^2(t) \\ \mathbf{x}^3(t) \\ \vdots \\ \mathbf{x}^m(t) \end{pmatrix}$$

motion space | *linearized motion* | *state space*

Motion texture

A graphical model representation



Motion textons

- A set of basic motion elements
- Representation: LDS

Dynamics model: $\mathbf{X}_{t+1} = \mathbf{A}_t \mathbf{X}_t + \mathbf{V}_t$

Observation model: $\mathbf{Y}_t = \mathbf{C}_t \mathbf{X}_t + \mathbf{W}_t$

$\Rightarrow$ $\boldsymbol{\theta} = (\mathbf{A}, \mathbf{C}, \mathbf{V}, \mathbf{W})$ model parameters

An auto-regressive moving average process (ARMA)

Linear dynamic system (LDS)

- Previous work
  - Tracking and gait recognition [Bregler'97, North'00, Pavlovic'00, Bissacco'01]
  - Video synthesis [Soatto'01, Fitzgibbon'01]
- Challenge for character motion synthesis
  - Captured motion is non-stationary
- Our solution
  - Segment-based stationary processes (LDS')

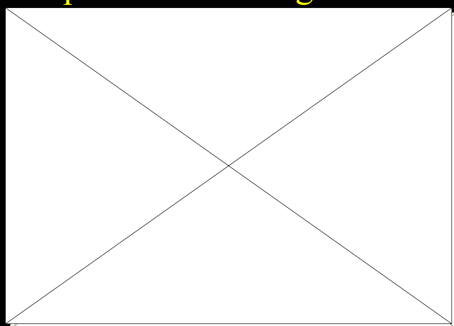
Learning motion texture

- Estimation by EM algorithm (segmentation!)
  - E-step: how many segments and where they are
  - M-step: fitting LDS parameters $\boldsymbol{\theta}$ for the segments labeled by the same texton
  - Transition graph (M): counting segment labels
- A maximum likelihood solution

$$\{\hat{\boldsymbol{\theta}}, \hat{\mathbf{M}}\} = \underset{\{\boldsymbol{\theta}, \mathbf{M}\}}{\mathbf{argmax}}\, \mathbf{P}(\mathbf{Y}_{1:T} \mid \boldsymbol{\theta}, \mathbf{M})$$

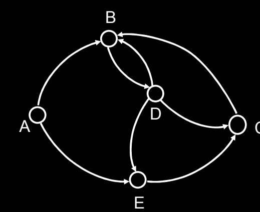
$(\boldsymbol{\theta} : \text{LDS parameters}, \quad \mathbf{M} : \text{transition graph})$

Unsupervised learning



With the texton distribution

- Random walk
  - ABDCBDEC……
- Multiple paths between any pairs
  - {ABDC}, {AEC}
- Add variations to the synthesized path
  - {A$\overline{\text{E}}$C}

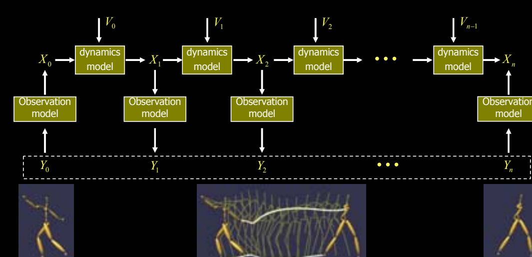


A two-step synthesis algorithm

- Texton path planning using DP
  - Finding the lowest cost path
  - Specifying the path length
- Synthesizing a single texton
  - Smooth transition between textons

Texton synthesis by constrained LDS

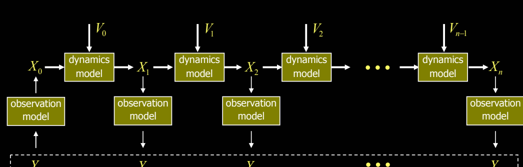
Sampling noise and incorporating both boundary constraints

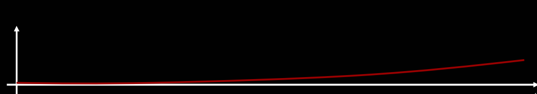


Solving a block-banded linear system

Texton synthesis by sampling noise

Without end constraints





Error accumulation

Video: synthesis and editing results

Video: synthesized dance motion

Conclusion

- **Motion texture**
  - Two levels: textons and distribution
  - An unsupervised learning algorithm
- **Real-time synthesis**
  - Texton path planning
  - Constrained synthesis
  - Generate fine details
  - Interactive editing

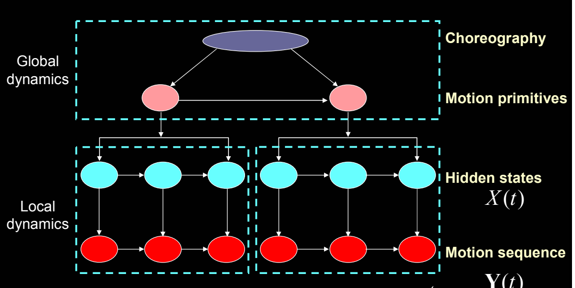
Acknowledgements

- **Dr. Ying-Qing Xu of MSRA**
- **Dr. Michael Cohen of MSR**

Thanks!

hshum@microsoft.com

An example: dance motion

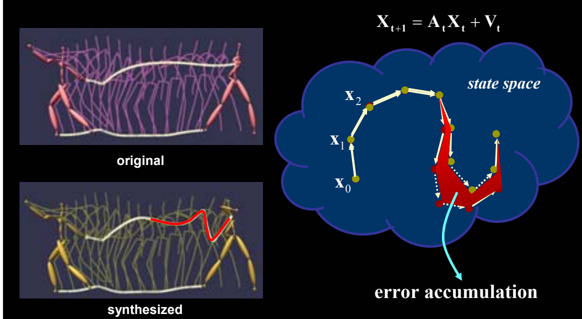


Problem

- We assume that motion textons are realizations of second-order stationary stochastic processes
- But in synthesis, the dynamics may deviate from the original one as time progresses
- Error accumulation in state space results in motion artifacts

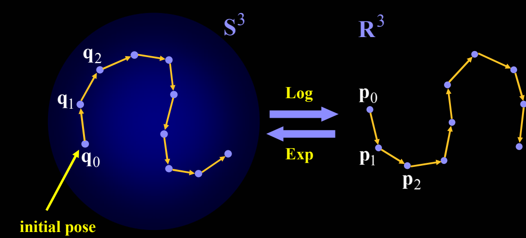
original *synthesized*

An illustration in the state space



Data transformation

Transformation between angular and linear signals



Learning motion texture

- A maximum likelihood solution:

  $$\{\hat{\theta}, \hat{M}\} = \underset{\{\theta, M\}}{\operatorname{argmax}} P(Y_{1:T} \mid \theta, M)$$

  $(\theta : \text{LDS' parameters}, \quad M : \text{transition matrix})$

- Introduce two hidden variables
  - *L* --- segment labels
  - *H* --- segment points
- Estimation by EM algorithm
  - E-step: infer *L* and *H*, and estimate optimal state sequence
  - M-step: update θ by fitting LDS', *M* by counting segment labels

Q/A: why not using derivative of rotation

- **In synthesis, we would like to constrain the end pose of the motion instead of the end angular speed of it.**
- **We are assuming that the identity in the exp map is the key pose.**
- **This assumption may result in singularity.**
- **Fortunately, the rotation change in each texton is small (< Pi).**

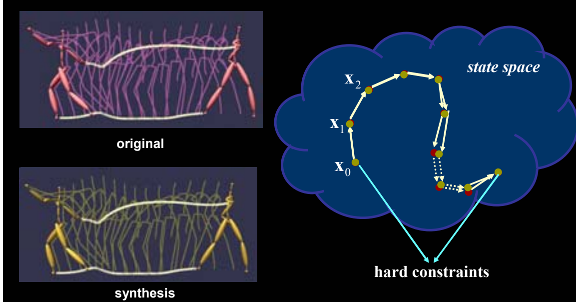
Q/A: SLDS and Motion Texture

- **We don't model each frame as a mixture of LDS'**
- **No approximate inference in our model**
- **Segment-based LDS models are desirable because they better capture the local consistent dynamics**

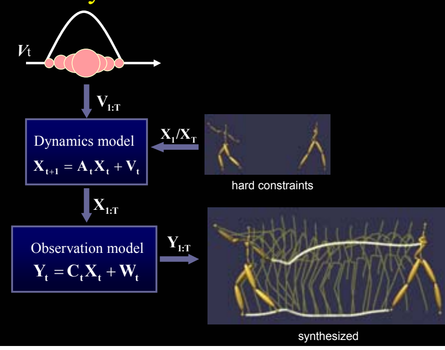
Q/A: Related work

- Motion texture
  [Bregler'00]
- Texton
  [Julesz'81, Malik'99, Zhu'02, Guo'01, Liang'01]
- Linear dynamic system
  [Bregler'97, Soatto'01, Fitzgibbon'01]
- Modeling nonlinear dynamics
  [Blake'02, Brand'00, Pavlovic'00]

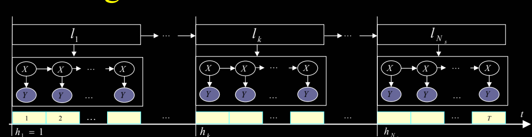
Texton synthesis with constrained LDS



Texton synthesis with constrained LDS



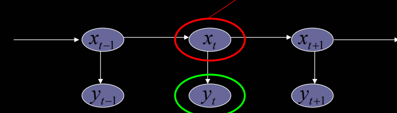
Learning Motion Texture



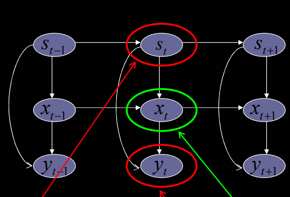
- A maximum likelihood solution: $\{\hat{\theta}, \hat{M}\} = \arg\max_{\{\theta, M\}} P(Y_{1:T} \mid \theta, M)$
- Introduce two hidden variables
  - *L* --- segment labels
  - *H* --- segment points
- Estimation by EM algorithm
  - E-step: infer *L* and *H*
  - M-step: update θ by fitting LDS', *M* by counting segment labels

Motion textons

Linear dynamic system



Switched LDS



Motion Textures

- State model: first-order Markov process

  $$P(s_t = j \mid s_{t-1} = i) = T_{ij}$$

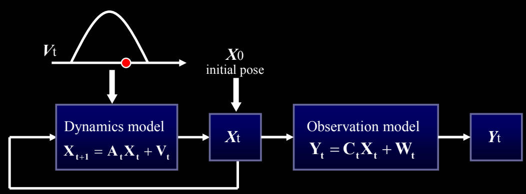
- Dynamics model

  $$P(x_t \mid x_{t-1}, s_t = i) = A_i x_{t-1} + v_i$$

- Observation model

  $$P(y_t \mid x_t, s_t = i) = C_i x_t + w_i$$

Texton synthesis by sampling noise

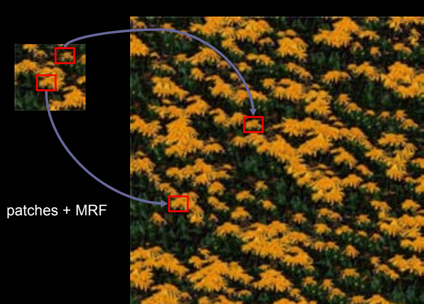


- Sample the initial pose
- Synthesize a frame by the observation model
- Draw samples from the noise term V_t
- Compute X_t by the dynamic model

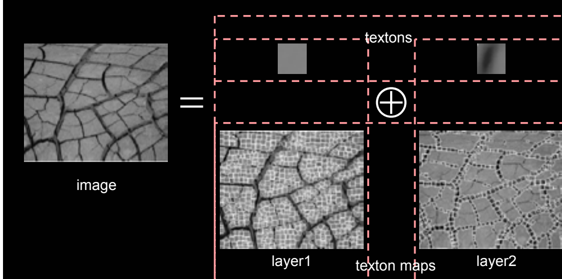
An analogy to 2D texture images

- An image texture: two-dimensional spatial distribution
- In psychology, basic texture elements are called "*texton*" or "*texel*" [Julesz'81]
- In early vision, natural visual patterns consist of *multiple layers of stochastic processes* [Marr'82]

Patch-based sampling (Liang et.al. '01)



Integrating descriptive and generative models -- (Guo et.al. '01)



But what are image textons?

Open question: what is a texton? ☹

Goal: model mo-cap data

- Generalize the stochastic and dynamic nature
- Synthesize realistic character motion
- Edit at both the high and low levels