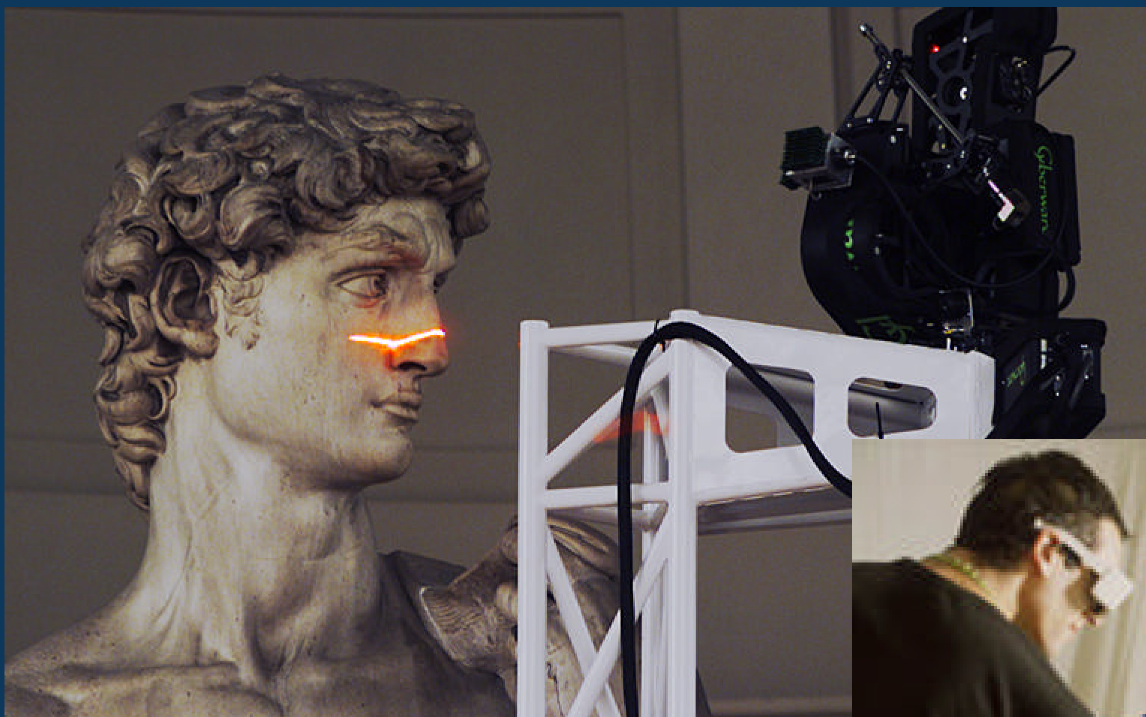
Bayesian Learning for Computer graphics

Aaron Hertzmann
University of Toronto

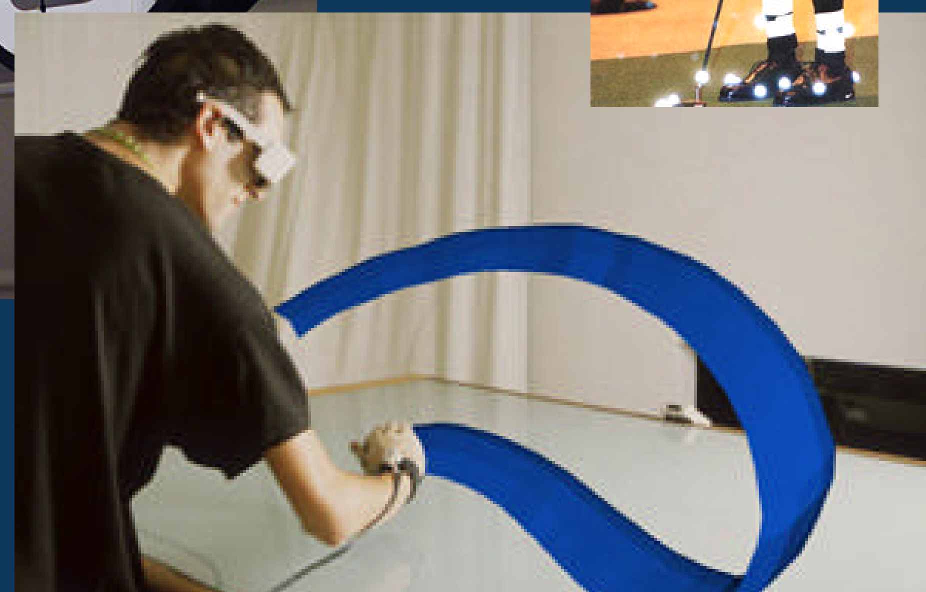
Computers are really fast

- If you can create it, you can render it

How do you create it?



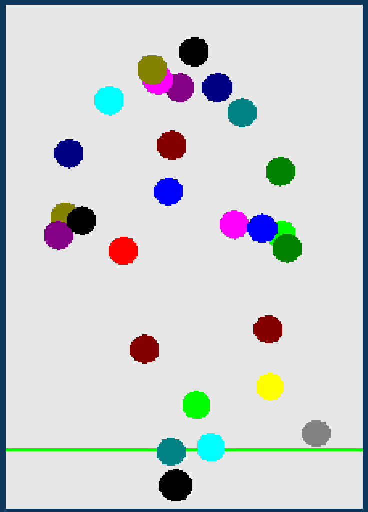
Digital Michaelangelo Project

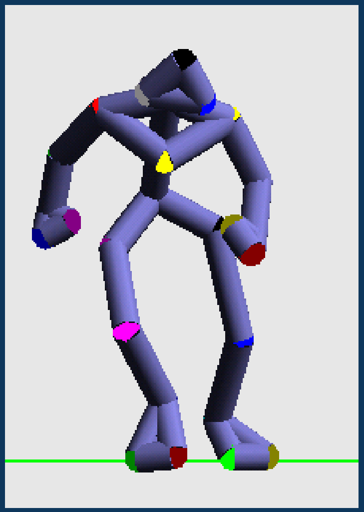


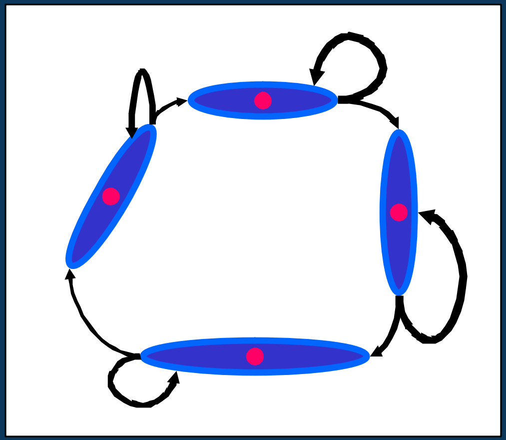
Steven Schkolne

Two problems

- How do we get the data into the computer?
- How do we manipulate it?

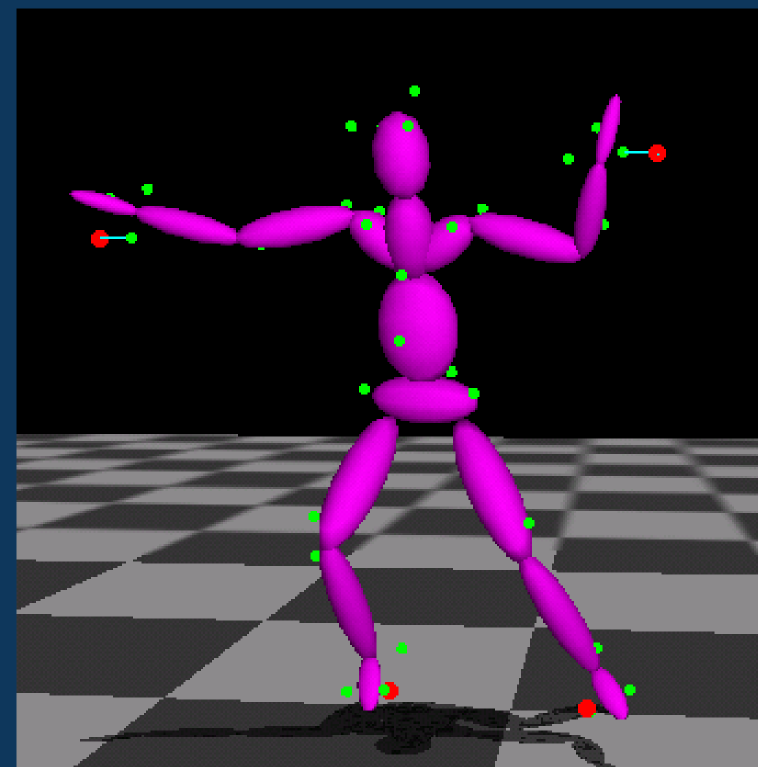






Data fitting

$$\sum_{j,t} \|m_j(q_t) - s_{j,t}\|^2$$
$$+ w_1 \sum_{j,t} \|q_t - q_{t-1}\|^2$$
$$+ w_2 \sum_{j,t} \|\ldots\|^2$$



Key questions

- How do you fit a model to data?
  - How do you choose weights and thresholds?
  - How do you incorporate prior knowledge?
  - How do you merge multiple sources of information?
  - How do you model uncertainty?

Bayesian reasoning provides a solution

Talk outline

- Bayesian reasoning
- Facial modeling example
- Non-rigid modeling from video

What is reasoning?

- How do people reason?
- How should computers do it?

Aristotelian Logic

- If **A** is true, then **B** is true
- **A** is true
- Therefore, **B** is true

A: My car was stolen
B: My car isn’t where I left it

Real-world is uncertain

Problems with pure logic

- Don’t have perfect information
- Don’t really know the model
- Model is non-deterministic

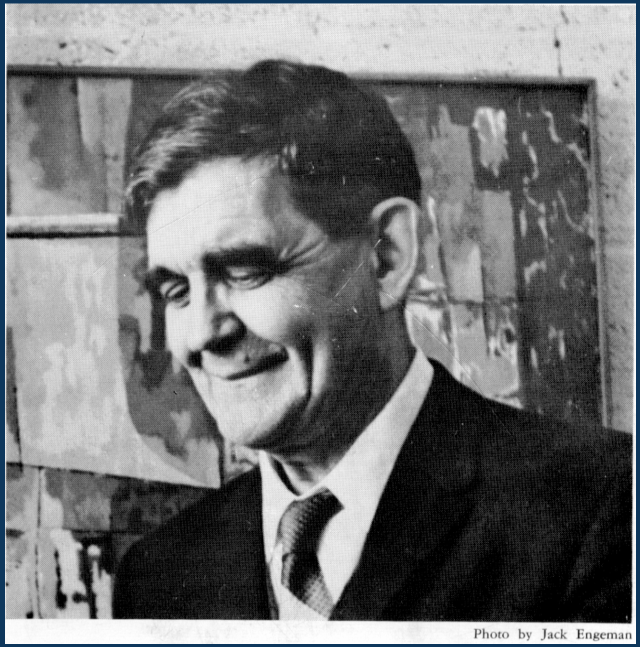
Beliefs

- Let B(X) = "belief in X",
- B(¬X) = "belief in not X"

1. An ordering of beliefs exists
2. $B(X) = f(B(\neg X))$
3. $B(X) = g(B(X|Y), B(Y))$

Cox axioms

R.T. Cox, “Probability, frequency, and reasonable expectation,” American J. Physics, 14(1):1-13, 1946



$$p(\mathrm{TRUE}) = 1$$

$$p(A) = \sum_{\{B=b_i\}} p(A, B)$$

$$p(A, B) = p(A|B)p(B)$$

“Probability theory is nothing more than common sense reduced to calculation.”

- Pierre-Simon Laplace, 1814

Bayesian vs. Frequentist

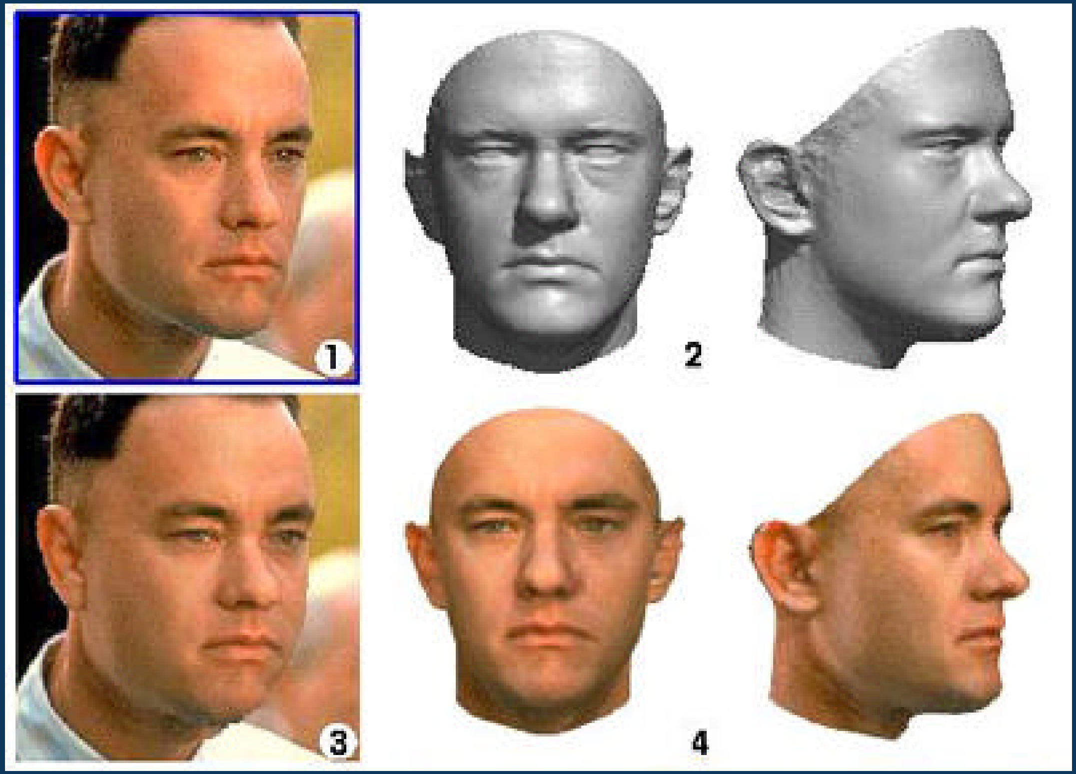
- Frequentist (“Orthodox”):
  - Probability = percentage of events in infinite trials
- Medicine, biology: Frequentist
- Astronomy, geology, EE, computer vision: largely Bayesian

Learning in a nutshell

- Create a mathematical model
- Get data
- Solve for unknowns

Face modeling

- Blanz, Vetter, “A Morphable Model for the Synthesis of 3D Faces,” SIGGRAPH 99



Generative model

- **Faces come from a Gaussian**

$$p(S|\bar{S};\Sigma) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} e^{-(S-\bar{S})^T\Sigma^{-1}(S-\bar{S})/2}$$

$$p(\{S_i\}|\bar{S};\Sigma) = \prod_i p(S_i|\bar{S};\Sigma)$$

Learning

$$\arg\max_{\bar{S},\Sigma}\ p(\bar{S};\Sigma|\{S_i\})$$

S_i

Bayes Rule

$$P(A, B) = p(A|B)p(B)$$
$$= p(B|A)p(A)$$

$$P(B|A) = p(A|B)p(B)/p(A)$$

$$p(model|data) \propto p(data|model)p(model)$$

Often: $p(model|data) \propto p(data|model)$

Learning a Gaussian

$$\arg\max_{\mu,\Sigma} p(\mu, \Sigma \mid \{S_i\})$$

$$= \arg\max_{\mu,\Sigma} p(\{S_i\} \mid \mu, \Sigma)$$

$$= \arg\max_{\mu,\Sigma} \prod_i p(S_i \mid \mu, \Sigma)$$

$$= \arg\max_{\mu,\Sigma} \prod_i \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{-\frac{1}{2}(S_i - \mu)^T \Sigma^{-1} (S_i - \mu)/2}$$

$$\mu \qquad \sum_i S_i / N$$

$$\Sigma \qquad \sum_i (S_i - \mu)(S_i - \mu)^T / N$$

Maximization trick

- Maximize $p(x)$

  <-> minimize à $\ln p(x)$

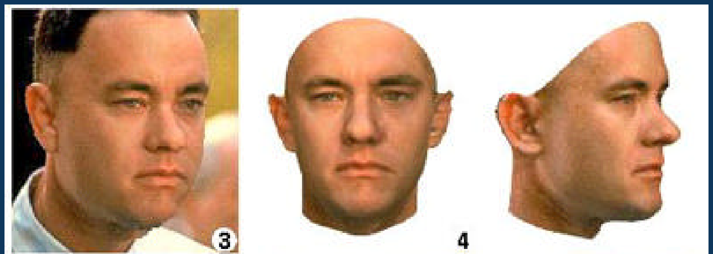
Fitting a face to an image

Generative model

$$p(S|\bar{S};\Sigma) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} e^{-(S-\bar{S})^T\Sigma^{-1}(S-\bar{S})/2}$$

$$I = \text{Render}(S;\theta) + n$$

$$\Rightarrow p(I|S;\theta;\sigma^2) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} e^{-||I-\text{Render}(S;\theta)||^2/2\sigma^2}$$



Fitting a face to an image

Maximize

$$p(S, \alpha \mid I, \bar{S}, \Sigma, \sigma^2)$$

minimize

$$-\ln p(S, \alpha \mid I, \bar{S}, \Sigma, \sigma^2) =$$

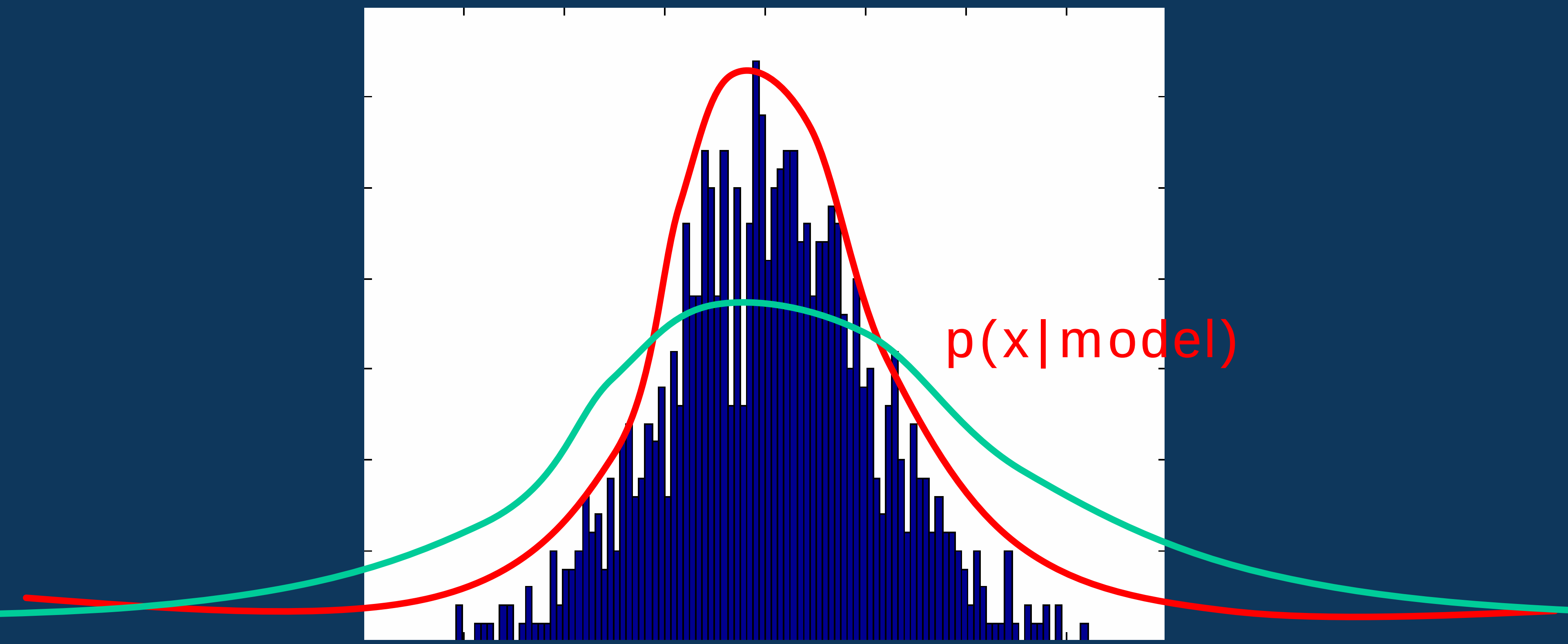
$$\|I - \mathrm{Render}(S, \alpha)\|^2 / 2\sigma^2 + (S - \bar{S})^T \Sigma^{-1} (S - \bar{S}) / 2$$

$$+ \frac{N}{2} \ln 2\pi\sigma^2 + \frac{1}{2} \ln (2\pi)^d |\Sigma|$$

Why does it work?

$$p(\{x_i\}|\text{model}) = \prod_i p(x_i|\text{model})$$

$$\text{s.t.} \sum_x p(x|\text{model}) = 1$$



General features

- Models uncertainty
- Applies to any generative model
- Merge multiple sources of information
- Learn *all* the parameters

Caveats

- Still need to understand the model
- Not necessarily tractable
- Potentially more involved than *ad hoc* methods

Applications in graphics

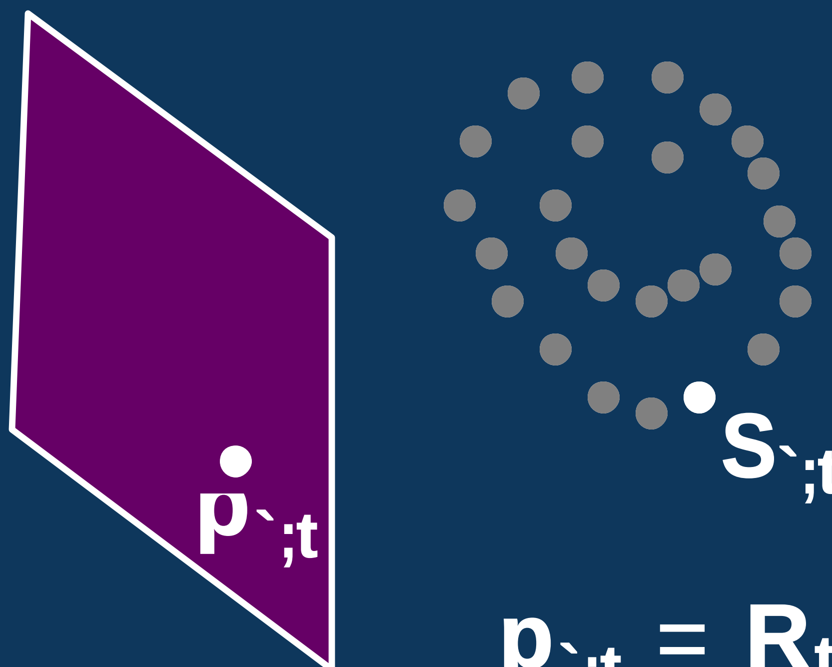
- Shape and motion capture
- Learning styles and generating new data

Learning Non-Rigid 3D Shape from 2D Motion

Joint work with Lorenzo Torresani and Chris Bregler (Stanford, NYU)

Camera geometry

Orthographic projection



$$p_{`;t} = R_t S_{`;t} + T_t$$

Non-rigid reconstruction

Input: 2D point tracks

Output: 3D nonrigid motion

$$p_t = R_t S_t + T_t$$

Totally ambiguous!

Shape reconstruction

Least-squares version

$$p_t = R_t S_t + T_t$$

minimize

$$\sum_t ||p_t - R_t S_t - T_t||^2 + w \sum_t ||S_t - \bar{S}||^2$$

Bayesian formulation

$$p(S_t|\bar{S}, \Sigma) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} e^{-(S_t-\bar{S})^T\Sigma^{-1}(S_t-\bar{S})/2}$$

$$p_{\lambda,t} = R_t S_{\lambda,t} + T_t + n \qquad n \sim N(0, \sigma^2)$$

---

maximize $p(R, S, T, \bar{S}, \Sigma | P)$

$$\propto p(P|R, S, T)\, p(S|\bar{S}, \Sigma)$$

How does it work?

Maximize $p(R, S, T, \bar{S}, \Sigma \mid P)$

$$\propto p(P \mid R, S, T)\, p(S \mid \bar{S}, \Sigma)$$

Minimize

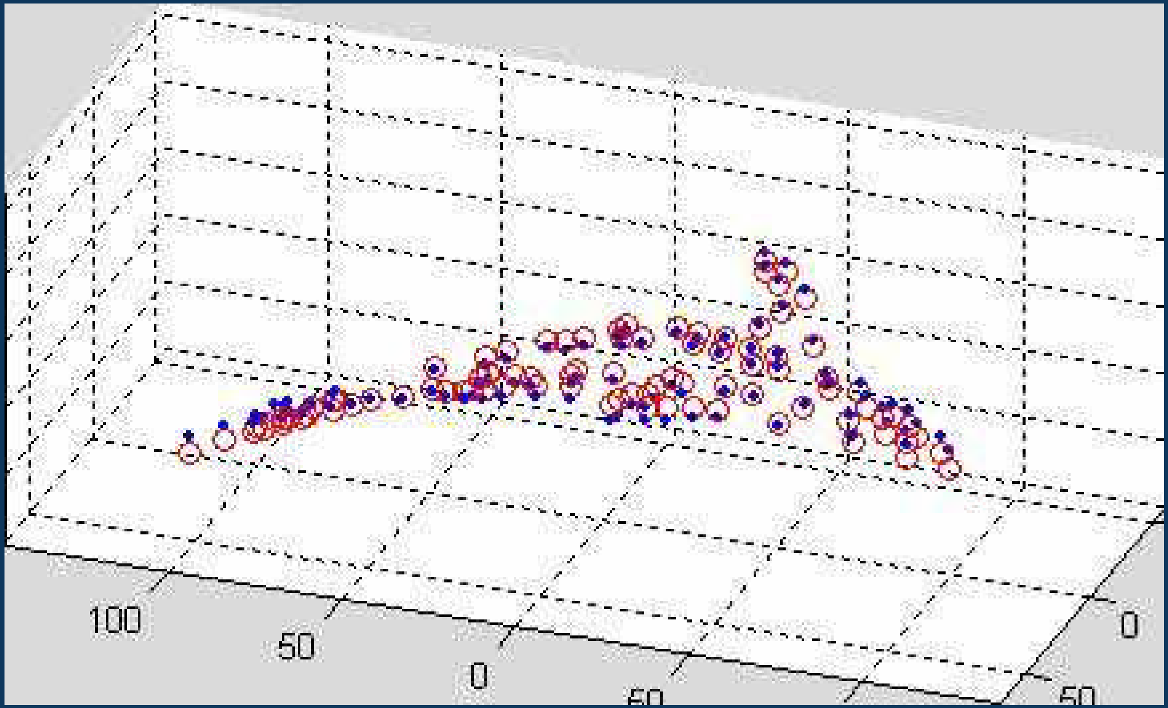
$$-\sum_t \ln p(P_t \mid R_t, S_t, T_t) - \sum_t \ln p(S_t \mid \bar{S}, \Sigma)$$

$$= \sum_t \| p_t - R_t S_t - T_t \|^2 / 2\sigma^2 + \frac{N}{2} \ln \sigma^2$$

$$+ \sum_t \frac{1}{2} (S_t - \bar{S})^T \Sigma^{-1} (S_t - \bar{S}) + \frac{T}{2} \ln |\Sigma|$$

(actual system is slightly more sophisticated)

Results



Input

Least-squares, no reg.

View 2

Least-squares, reg.

Gaussian

LDS

Conclusions

- Bayesian methods provide unified framework
- Build a model, and reason about it
- The future of data-driven graphics