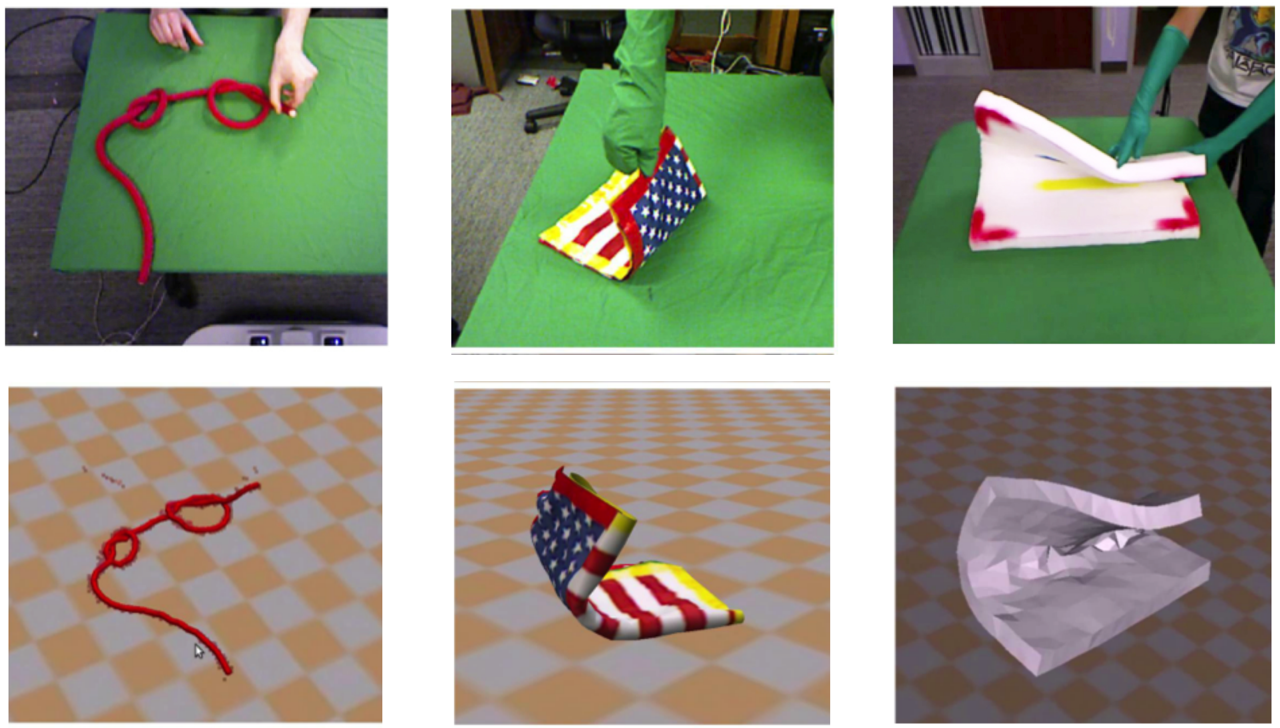
Tracking Deformable Objects with Point Clouds

John Schulman, Alex Lee, Jonathan Ho, Pieter Abbeel

UC Berkeley, EECS Department

Goal

- Track deformable objects from point cloud data
- Assumption: we have a physical model of the object



Kinect RGB

Rendering of state estimate

1D 2D 3D

Energy Minimization Methods

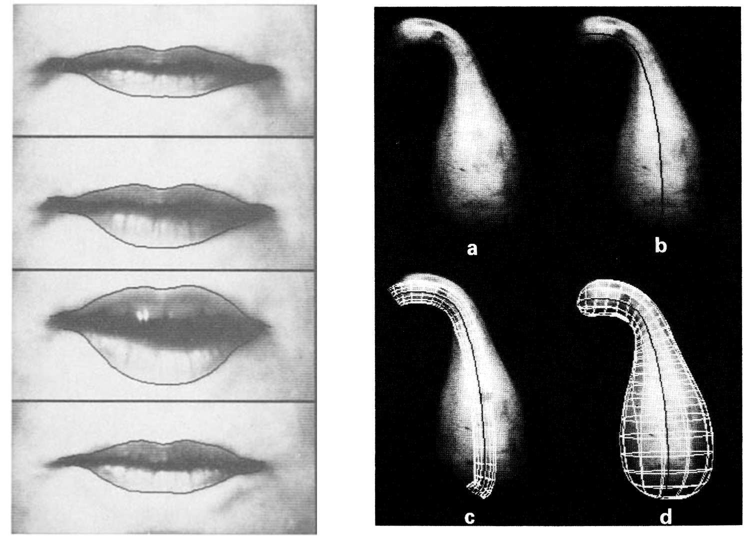
x: state estimate **y**: observation

$$E_{\text{total}}(\mathbf{x}, \mathbf{y}) = E_{\text{internal}}(\mathbf{x}) + E_{\text{external}}(\mathbf{x}, \mathbf{y})$$

discourage bending and stretching

encourage model to match up with image

$$\min_{\mathbf{x}} E_{\text{total}}(\mathbf{x}, \mathbf{y})$$



Kass, Terzopoulos, Witkin, 1988

Energy Minimization Methods

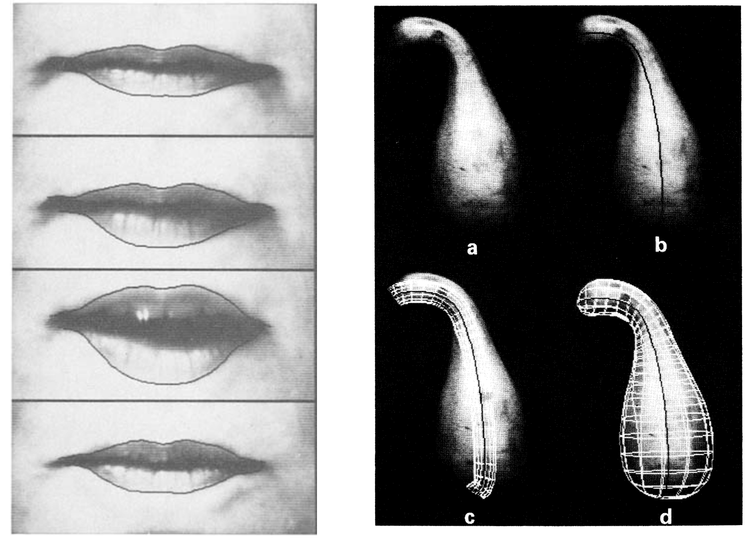
x: state estimate   **y**: observation

$$E_{\text{total}}(\mathbf{x}, \mathbf{y}) = E_{\text{internal}}(\mathbf{x}) + E_{\text{external}}(\mathbf{x}, \mathbf{y})$$

discourage bending and stretching

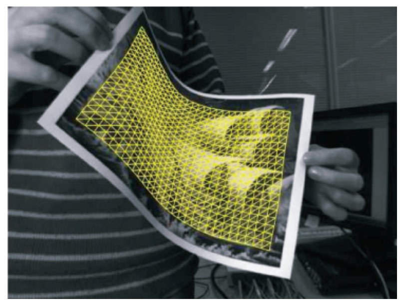
encourage model to match up with image

$$\min_{\mathbf{x}} E_{\text{total}}(\mathbf{x}, \mathbf{y})$$

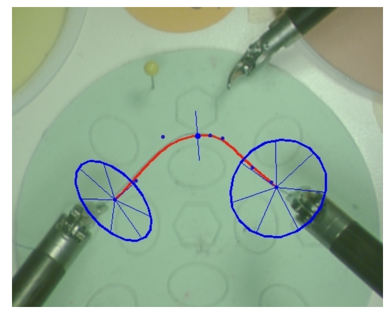


Kass, Terzopoulos, Witkin, 1988

Wuhrer, Lang, & Shu 2012



Saltzman et al. 2007



Padoy & Hager 2011

Probabilistic Methods

Probabilistic Methods

Energy minimization methods

$$E_{\text{total}}(\mathbf{x}, \mathbf{y}) = E_{\text{internal}}(\mathbf{x}) + E_{\text{external}}(\mathbf{x}, \mathbf{y})$$

$$\min_{\mathbf{x}} E_{\text{total}}(\mathbf{x}, \mathbf{y})$$

Probabilistic Methods

x: state estimate **y**: observation

$$p(\mathbf{x},\mathbf{y}) \propto e^{-E_{\text{internal}}(\mathbf{x})} e^{-E_{\text{external}}(\mathbf{x},\mathbf{y})}$$

$$\max_{\mathbf{x}} p(\mathbf{x},\mathbf{y})$$ (MAP estimation)

Energy minimization methods

$$E_{\text{total}}(\mathbf{x},\mathbf{y}) = E_{\text{internal}}(\mathbf{x}) + E_{\text{external}}(\mathbf{x},\mathbf{y})$$

$$\min_{\mathbf{x}} E_{\text{total}}(\mathbf{x},\mathbf{y})$$

Probabilistic Methods

x: state estimate **y**: observation

$$p(\mathbf{x}, \mathbf{y}) \propto e^{-E_{\text{internal}}(\mathbf{x})} e^{-E_{\text{external}}(\mathbf{x}, \mathbf{y})}$$

$$\max_{\mathbf{x}} p(\mathbf{x}, \mathbf{y})$$ (MAP estimation)

Energy minimization methods

$$E_{\text{total}}(\mathbf{x}, \mathbf{y}) = E_{\text{internal}}(\mathbf{x}) + E_{\text{external}}(\mathbf{x}, \mathbf{y})$$

$$\min_{\mathbf{x}} E_{\text{total}}(\mathbf{x}, \mathbf{y})$$

$$p(\mathbf{x}, \mathbf{y}) \propto \sum_{\mathbf{z}} e^{-E(\mathbf{x}, \mathbf{y}, \mathbf{z})}$$

z: correspondences

Probabilistic Methods

x: state estimate **y**: observation

$$p(\mathbf{x},\mathbf{y}) \propto e^{-E_{\text{internal}}(\mathbf{x})} e^{-E_{\text{external}}(\mathbf{x},\mathbf{y})}$$

$$\max_{\mathbf{x}} p(\mathbf{x},\mathbf{y})$$ (MAP estimation)

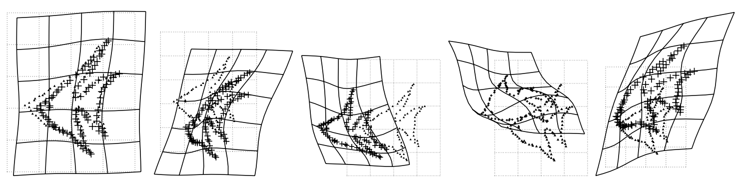
Energy minimization methods

$$E_{\text{total}}(\mathbf{x},\mathbf{y}) = E_{\text{internal}}(\mathbf{x}) + E_{\text{external}}(\mathbf{x},\mathbf{y})$$

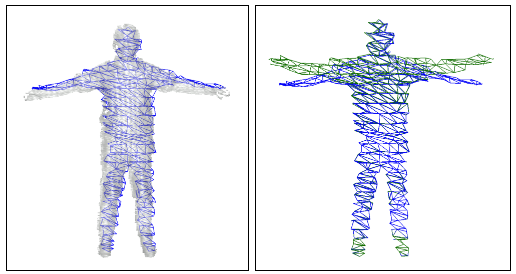
$$\min_{\mathbf{x}} E_{\text{total}}(\mathbf{x},\mathbf{y})$$

$$p(\mathbf{x},\mathbf{y}) \propto \sum_{\mathbf{z}} e^{-E(\mathbf{x},\mathbf{y},\mathbf{z})}$$

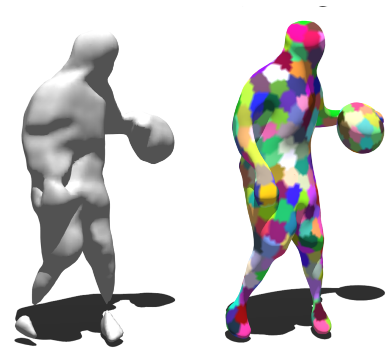
z: correspondences



Myronenko & Song 2007



Hahnel, Thrun & Burgard 2003



Cagniart, Boyer, & Ilic 2010

Challenges

Challenges

- Observation modeling:
  - correspondence problem
  - noise
  - occlusions

Challenges

- Observation modeling:
  - correspondence problem
  - noise
  - occlusions
- Physical constraints:
  - non-penetration
  - hard constraints on bending and stretching

Challenges Contributions

- Observation modeling:
  - correspondence problem
  - noise
  - occlusions
- Physical constraints:
  - non-penetration
  - hard constraints on bending and stretching

Challenges Contributions

- Observation modeling:
  - correspondence problem
  - noise
  - occlusions
- Physical constraints:
  - non-penetration
  - hard constraints on bending and stretching

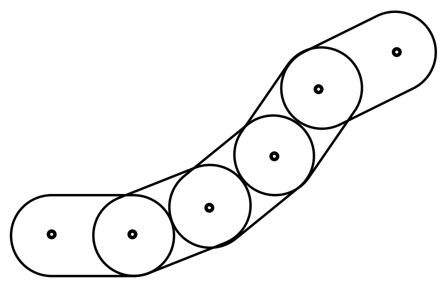
- Modeling contribution:
  - Probabilistic model that captures correspondence problem, noise, and occlusions

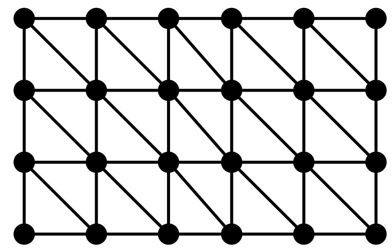
Challenges Contributions

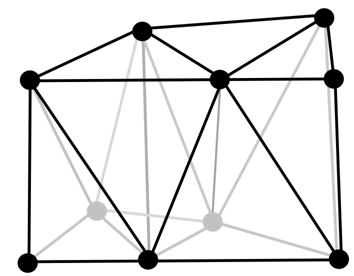
- Observation modeling:
  - correspondence problem
  - noise
  - occlusions
- Physical constraints:
  - non-penetration
  - hard constraints on bending and stretching

- Modeling contribution:
  - Probabilistic model that captures correspondence problem, noise, and occlusions
- Algorithmic contribution:
  - Modification of the EM algorithm that accounts for physical constraints
  - Operates by only introducing external forces into physics simulation engines

Preliminaries

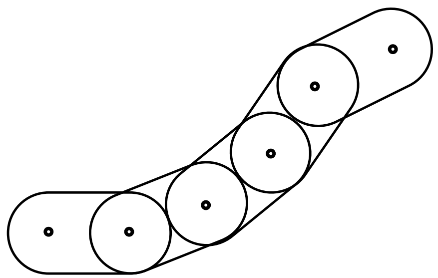


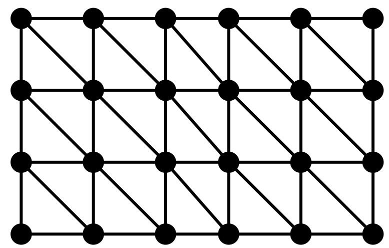


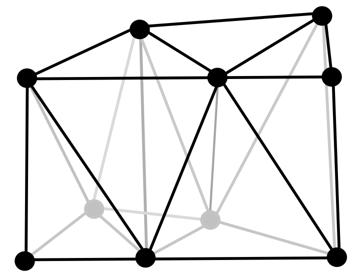


Preliminaries

- Physical models

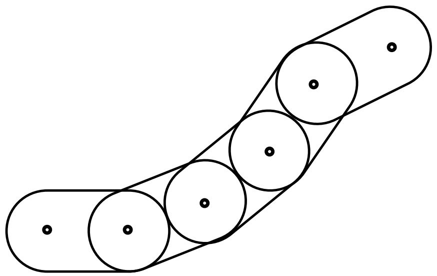


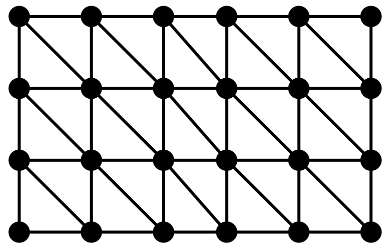


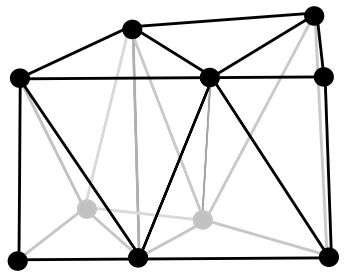


Preliminaries

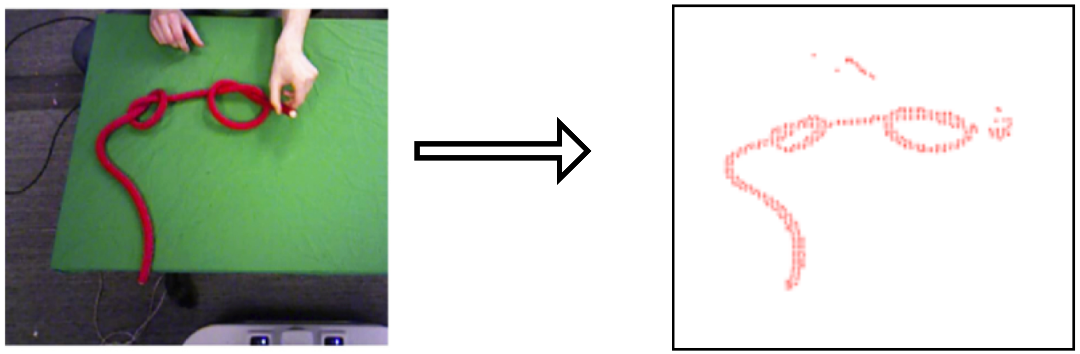
- Physical models



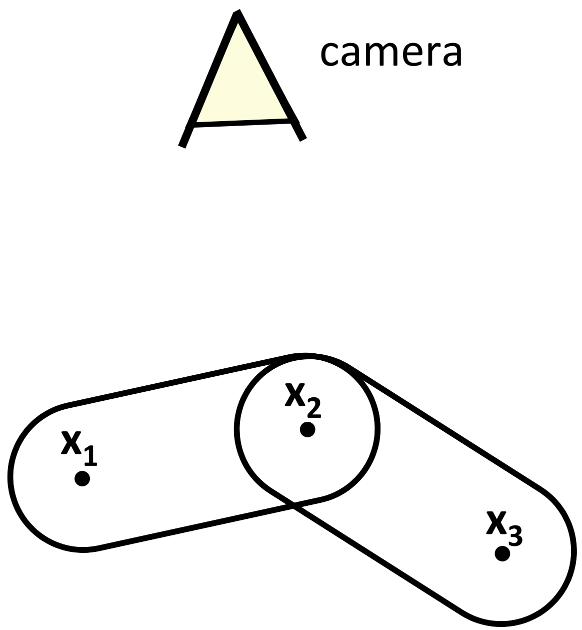




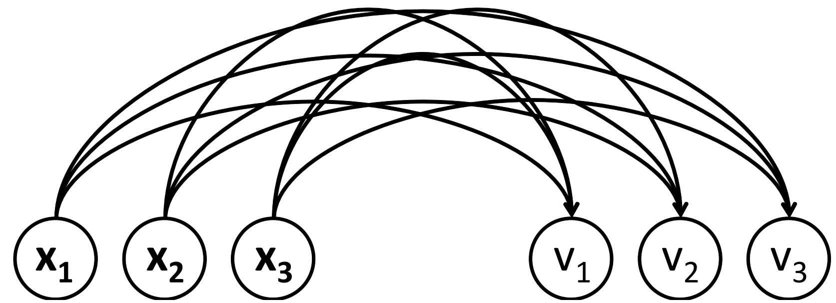
- Image / point cloud processing
  - background subtraction (may have false positives & negatives)



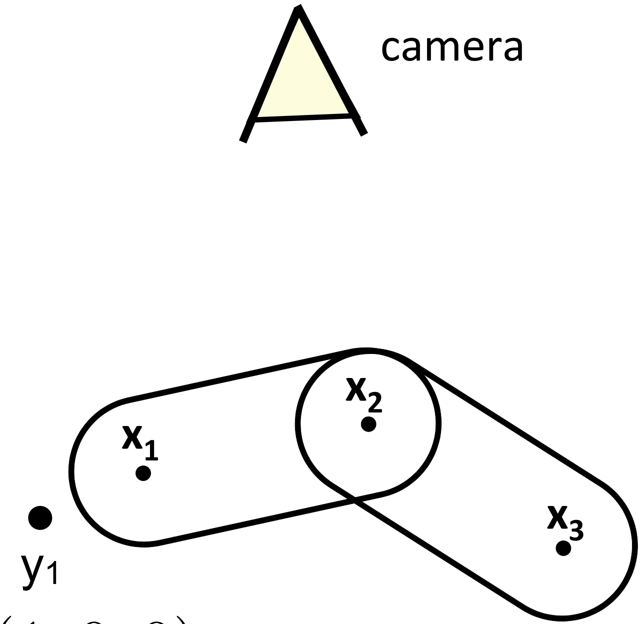
Observation Model



Graphical model

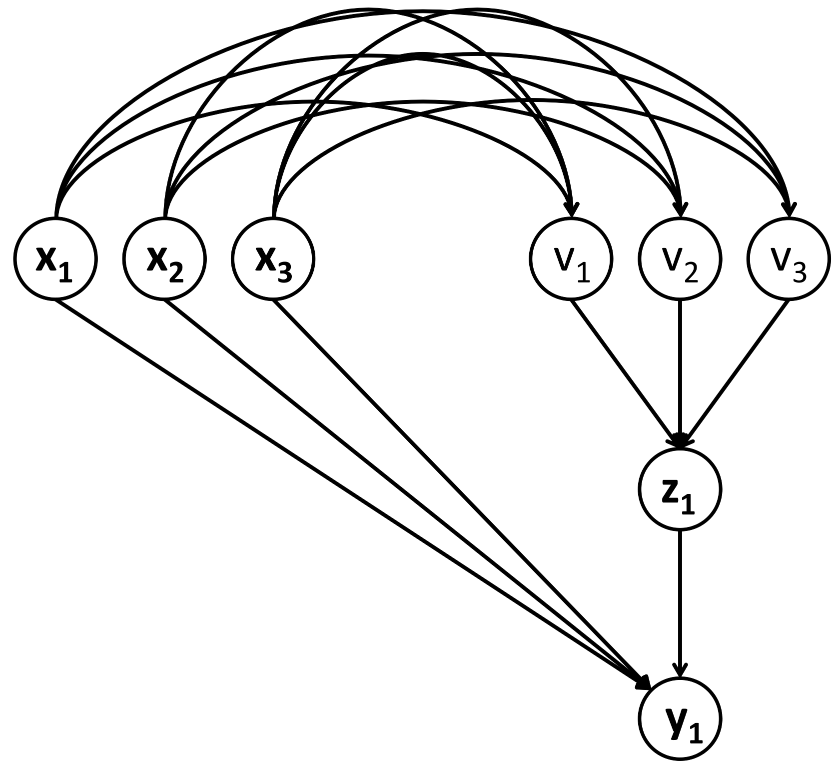


Observation Model

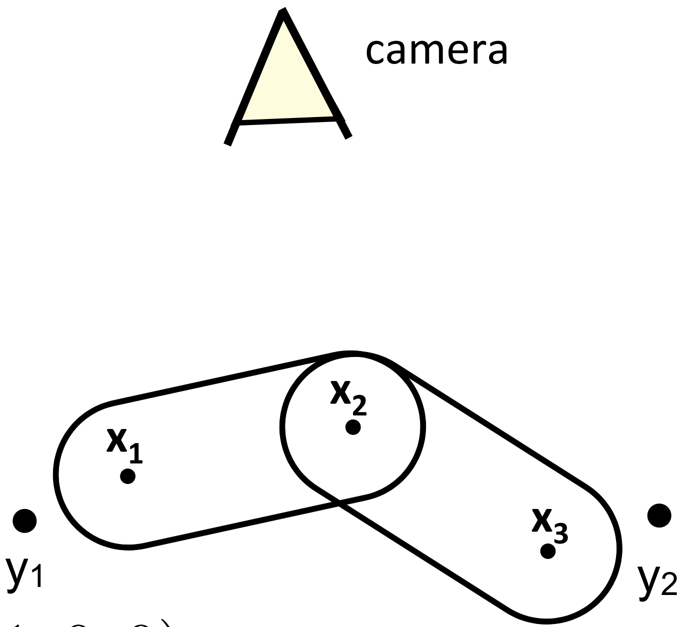


$$z_1 = (1, 0, 0)$$
$$y_1 = \mathcal{N}(\mathbf{x}_1, \sigma^2)$$

Graphical model



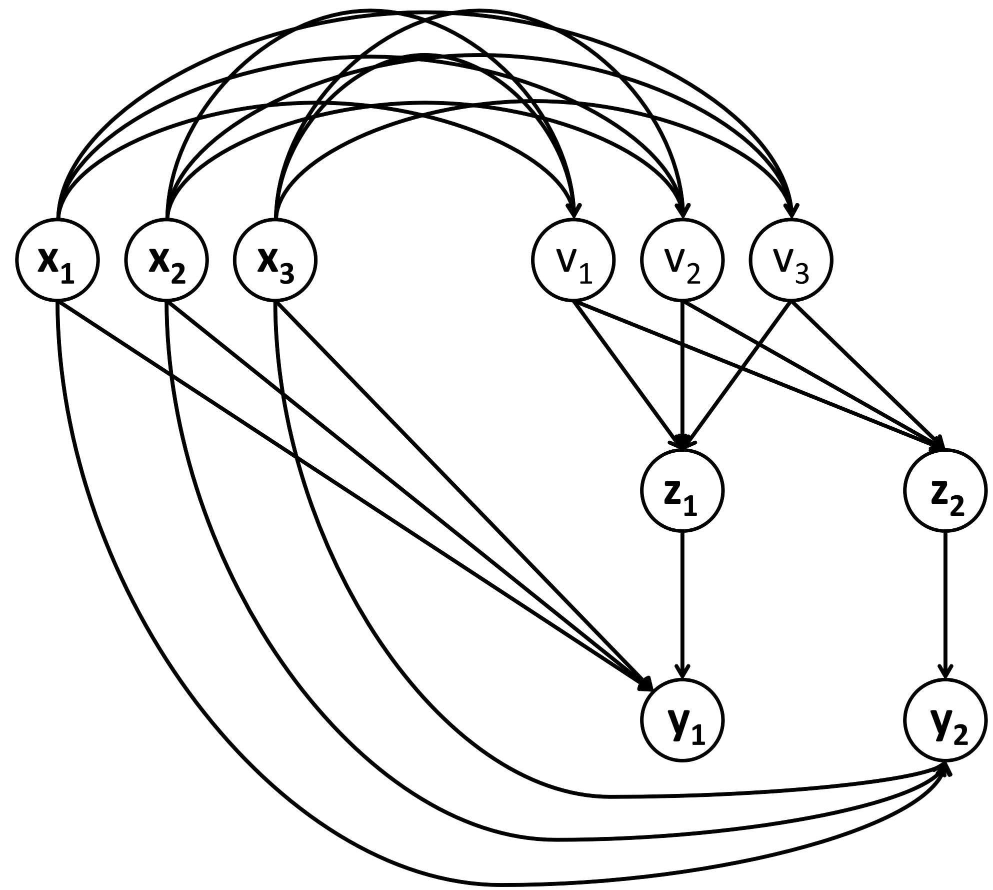
Observation Model



$$z_1 = (1, 0, 0)$$
$$y_1 = \mathcal{N}(\mathbf{x}_1, \sigma^2)$$

$$z_2 = (0, 0, 1)$$
$$y_2 = \mathcal{N}(\mathbf{x}_3, \sigma^2)$$

Graphical model



EM Algorithm

EM Algorithm

- MAP inference problem:

$$\arg\max_{\mathbf{x}} \log p(\mathbf{x}, \mathbf{y}) = \arg\max_{\mathbf{x}} \log \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{y}, \mathbf{z})$$

EM Algorithm

- MAP inference problem:

$$\arg\max_{\mathbf{x}} \log p(\mathbf{x}, \mathbf{y}) = \arg\max_{\mathbf{x}} \log \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{y}, \mathbf{z})$$

- For i=1,2,3,...

EM Algorithm

- MAP inference problem:

$$\arg\max_{\mathbf{x}} \log p(\mathbf{x}, \mathbf{y}) = \arg\max_{\mathbf{x}} \log \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{y}, \mathbf{z})$$

- For i=1,2,3,...

E Step: $q^{(i)}(\mathbf{z}) = p(\mathbf{z}|\mathbf{x}^{(i-1)}, \mathbf{y})$

Calculate posterior of latent variables

EM Algorithm

- MAP inference problem:

$$\arg\max_{\mathbf{x}} \log p(\mathbf{x}, \mathbf{y}) = \arg\max_{\mathbf{x}} \log \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{y}, \mathbf{z})$$

- For i=1,2,3,...

E Step: $$q^{(i)}(\mathbf{z}) = p(\mathbf{z}|\mathbf{x}^{(i-1)}, \mathbf{y})$$

Calculate posterior of latent variables

M Step: $$\mathbf{x}^{(i)} = \arg\max_{\mathbf{x}} A_{q^{(i)}}(\mathbf{x})$$

Maximize a lower bound to log-probability

$$A_{q^{(i)}}(\mathbf{x}) = \sum_{\mathbf{z}} q^{(i)}(\mathbf{z}) \log p(\mathbf{x}, \mathbf{y}, \mathbf{z}) + S(q^{(i)}) \leq \log p(\mathbf{x}, \mathbf{y})$$

Modified M Step

Modified M Step

- Standard M step (i^{th} iteration)

$$\mathbf{x}^{(i)} = \arg\max_{\mathbf{x}} A_{q^{(i)}}(\mathbf{x})$$

Modified M Step

- Standard M step (i[th] iteration)

$$\mathbf{x}^{(i)} = \arg\max_{\mathbf{x}} A_{q^{(i)}}(\mathbf{x})$$

- M step with physical constraints:
  - Repeatedly apply forces in gradient direction (+ damping) until convergence

$$\mathbf{f}_k = \nabla_{\mathbf{x}_k} A_q(\mathbf{x}) - \beta\dot{\mathbf{x}}$$

Modified M Step

- Standard M step (i^th iteration)

$$\mathbf{x}^{(i)} = \arg\max_{\mathbf{x}} A_{q^{(i)}}(\mathbf{x})$$

- M step with physical constraints:
  - Repeatedly apply forces in gradient direction (+ damping) until convergence

$$\mathbf{f}_k = \nabla_{\mathbf{x}_k} A_q(\mathbf{x}) - \beta \dot{\mathbf{x}}$$

  - physical interpretation:
    - simulating a system where -$A_{q^{(i)}}(\mathbf{x})$ is the potential energy,
    - damped physical system converges to local minimum of potential energy

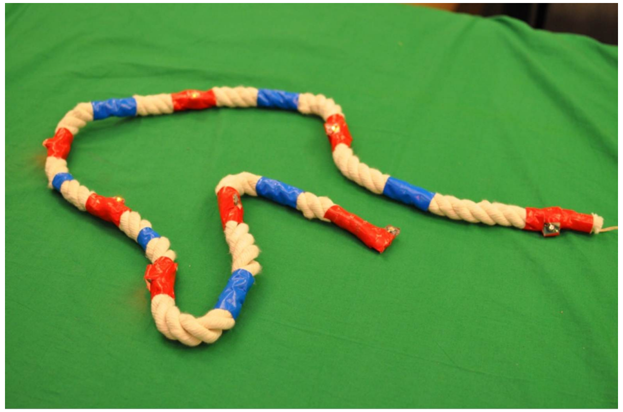
Real-time Implementation

Real-time Implementation

- For time t=1,2,...
  - Iterate until next point cloud received:
    - E Step
    - M Step

Demo video

Quantitative Evaluation of Accuracy



Motion capture

Experimental results

Experimental results

- 14 different tasks
  - human manipulates rope
  - robot manipulates rope
  - human manipulates cloth

Experimental results

- 14 different tasks
  - human manipulates rope
  - robot manipulates rope
  - human manipulates cloth
- Succeeds on 13/14 tasks with 2.5cm mean error on successful runs

Summary

Summary

- Modeling contribution:
  - Probabilistic model that captures correspondence problem, noise, and occlusions

Summary

- Modeling contribution:
  - Probabilistic model that captures correspondence problem, noise, and occlusions
- Algorithmic contribution:
  - Modification of the EM algorithm that can account for physical constraints
  - Operates by only introducing external forces into physics simulation engines
    - Use your favorite physics engine!

Summary

- Modeling contribution:
  - Probabilistic model that captures correspondence problem, noise, and occlusions
- Algorithmic contribution:
  - Modification of the EM algorithm that can account for physical constraints
  - Operates by only introducing external forces into physics simulation engines
    - Use your favorite physics engine!
- Implementation runs in real-time on standard computer

Thanks

Thanks

- Acknowledgements
  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE

- Code, data, paper:
  - http://rll.berkeley.edu/tracking

- Contact
  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements

  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE

- Code, data, paper:
  - http://rll.berkeley.edu/tracking
- Contact

  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements
  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE

- Code, data, paper:
  - http://rll.berkeley.edu/tracking

- Contact
  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements
  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE
- Code, data, paper:
  - http://rll.berkeley.edu/tracking
- Contact
  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements

  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE

- Code, data, paper:
  - http://rll.berkeley.edu/tracking
- Contact

  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements
  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE
- Code, data, paper:
  - http://rll.berkeley.edu/tracking
- Contact
  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements
  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE
- Code, data, paper:
  - http://rll.berkeley.edu/tracking
- Contact
  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements

  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE

- Code, data, paper:
  - http://rll.berkeley.edu/tracking
- Contact

  - John Schulman: joschu@eecs.berkeley.edu

Thanks

- Acknowledgements
  - Financial support: Intel, AFOSR-YIP
  - Open-source software: Bullet, PCL, OpenCV, ROS, OpenRAVE

- Code, data, paper:
  - http://rll.berkeley.edu/tracking

- Contact
  - John Schulman: joschu@eecs.berkeley.edu