Human Motion Analysis
Lecture 12: Discriminative Prediction II

Raquel Urtasun

TTI Chicago

May 31, 2010

Materials used for this lecture

- The slides for Non-parametric BP come from Erik Sudderth 2010 class on learning and inference on graphical models. Thanks Erik!
- See references for the rest of the class.

What did we look into last class?

- Local and global image features
- Similarities between images
- Discriminative prediction
  - NN
  - Regression
  - Mixture of experts

What are we going to see today?

Continue on discriminative prediction:

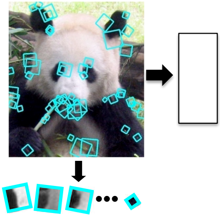
- Latent spaces for discriminative prediction
- Structure prediction

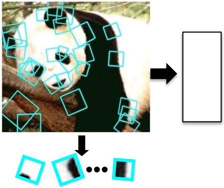
Look into combinations of generative and discriminative methods

No time for activity recognition: modern approaches are similar to object recognition.

Feature types

- Global vs local
- For local features: Interest points vs dense local features





Different descriptors

- Global descriptors: HOG, PHOG, Shape Context, GIST, HMAX
- Local descriptors: SIFT, SURF, Geometric Blur

Distances between features

- Global descriptors: euclidean, mahalanobis, histogram intersection
- Local features: BOW, matching, PMK, Spatial pyramid.
- Multiple Kernel Learning

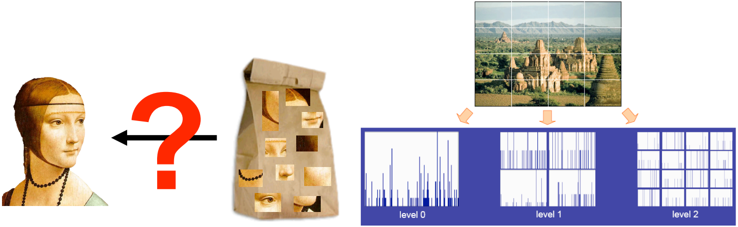


Figure: (left) BOW, (right) Spatial pyramid

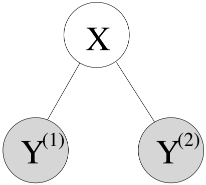
Discriminative approaches

- NN techniques: Linear search, Space partitioning (e.g., KD-trees), LSH, PSH.
- Regression: least-square regression, ridge regression, lasso, GP regression
- Mixture of experts due to multimodal mappings, e.g., mixtures of local GPs.

Shared latent space models

Many different models:

- Canonical Correlation Analysis (CCA).
- Shared-GPLVM (Shon et al. NIPS'06, Ek et al. MLMI'07, Navaratnam et al. ICCV'07).
- Shared-KIE (Sigal et al. CVPR'09).



They are effective when the views are correlated.

Canonical Correlation Analysis (CCA)

- Seek vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ so that the random variables $\mathbf{w}_1\mathbf{Y}^{(1)}$ and $\mathbf{w}_2\mathbf{Y}^{(2)}$ are maximally correlated

$$\rho = \frac{\mathbf{w}_1^T \Sigma_{12} \mathbf{w}_2}{\sqrt{\mathbf{w}_1^T \Sigma_{11} \mathbf{w}_1}\sqrt{\mathbf{w}_2^T \Sigma_{22} \mathbf{w}_2}}$$

- Using a change of basis $\mathbf{v}_1 = (\Sigma_{11})^{\frac{1}{2}}\mathbf{w}_1$ and $\mathbf{v}_2 = (\Sigma_{22})^{\frac{1}{2}}\mathbf{w}_2$ we can write

$$\rho = \frac{\mathbf{v}_1^T (\Sigma_{11})^{-\frac{1}{2}} \Sigma_{12} (\Sigma_{22})^{-\frac{1}{2}} \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \mathbf{v}_1}\sqrt{\mathbf{v}_2^T \mathbf{v}_2}}$$

Canonical Correlation Analysis (CCA)

- Seek vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ so that the random variables $\mathbf{w}_1\mathbf{Y}^{(1)}$ and $\mathbf{w}_2\mathbf{Y}^{(2)}$ are maximally correlated

$$\rho = \frac{\mathbf{w}_1^T \Sigma_{12} \mathbf{w}_2}{\sqrt{\mathbf{w}_1^T \Sigma_{11} \mathbf{w}_1}\sqrt{\mathbf{w}_2^T \Sigma_{22} \mathbf{w}_2}}$$

- Using a change of basis $\mathbf{v}_1 = (\Sigma_{11})^{\frac{1}{2}}\mathbf{w}_1$ and $\mathbf{v}_2 = (\Sigma_{22})^{\frac{1}{2}}\mathbf{w}_2$ we can write

$$\rho = \frac{\mathbf{v}_1^T (\Sigma_{11})^{-\frac{1}{2}} \Sigma_{12} (\Sigma_{22})^{-\frac{1}{2}} \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \mathbf{v}_1}\sqrt{\mathbf{v}_2^T \mathbf{v}_2}}$$

- Closed form solution: The maximum correlation is attained if $\mathbf{v}_1$ is the eigenvector with maximum eigenvalue of the matrix $(\Sigma_{11})^{-\frac{1}{2}}\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}(\Sigma_{11})^{-\frac{1}{2}}$.
- The subsequent pairs are found by using eigenvalues of decreasing magnitudes.
- Orthogonality is guaranteed by the symmetry of the correlation matrices.

Canonical Correlation Analysis (CCA)

- Seek vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ so that the random variables $\mathbf{w}_1\mathbf{Y}^{(1)}$ and $\mathbf{w}_2\mathbf{Y}^{(2)}$ are maximally correlated

$$\rho = \frac{\mathbf{w}_1^T \Sigma_{12} \mathbf{w}_2}{\sqrt{\mathbf{w}_1^T \Sigma_{11} \mathbf{w}_1}\sqrt{\mathbf{w}_2^T \Sigma_{22} \mathbf{w}_2}}$$

- Using a change of basis $\mathbf{v}_1 = (\Sigma_{11})^{\frac{1}{2}}\mathbf{w}_1$ and $\mathbf{v}_2 = (\Sigma_{22})^{\frac{1}{2}}\mathbf{w}_2$ we can write

$$\rho = \frac{\mathbf{v}_1^T (\Sigma_{11})^{-\frac{1}{2}} \Sigma_{12} (\Sigma_{22})^{-\frac{1}{2}} \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \mathbf{v}_1}\sqrt{\mathbf{v}_2^T \mathbf{v}_2}}$$

- Closed form solution: The maximum correlation is attained if $\mathbf{v}_1$ is the eigenvector with maximum eigenvalue of the matrix $(\Sigma_{11})^{-\frac{1}{2}} \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21} (\Sigma_{11})^{-\frac{1}{2}}$.
- The subsequent pairs are found by using eigenvalues of decreasing magnitudes.
- Orthogonality is guaranteed by the symmetry of the correlation matrices.

Some remarks on CCA

- Use the kernel trick to learn non-linear mappings Kernel CCA
- Problems with correlated noise
- Kernel CCA very sensitive to parameter tuning.

Shared Gaussian process latent variable model

- Model the mapping from a joint latent space to an observation spaces as

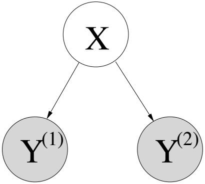
$$p(\mathbf{Y}^{(i)}|\mathbf{Z}^{(i)},\mathbf{X}) = \prod_{d=1}^{D_i} \mathcal{N}(\mathbf{Y}^{(i)}_{:,d}|0,\mathbf{K}^{(i)})$$

  where $\mathbf{K}^{(i)}$ is an $N \times N$ kernel matrix.

- The model is learn by minimizing the negative log likelihood

$$L_{data} = \sum_{i=1}^{V} \left(\frac{D_i}{2} \ln|\mathbf{K}^{(i)}| + \frac{D_i}{2} tr\left[(\mathbf{K}^{(i)})^{-1} \mathbf{Y}^{(i)} (\mathbf{Y}^{(i)})^T \right] \right) .$$

- For inference, the mean prediction from a joint latent coordinate to a view is given by $\bar{\mathbf{y}}^{(i)}_* = (\mathbf{k}^{(i)}_*)^T (\mathbf{K}^{(i)})^{-1} \mathbf{Y}^{(i)}$.



Shared GPLVM

- Developed by Shon et la. 06.
- Adapted by Ek et al. 07 and Navaratnam et al. 07 to solve pose estimation.

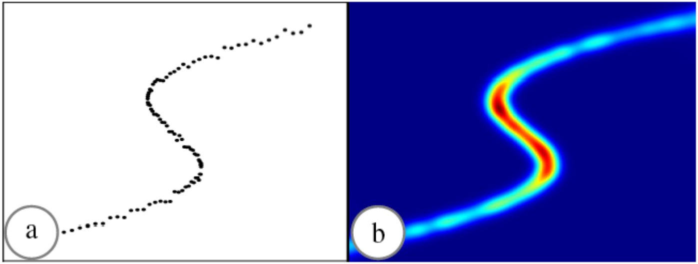
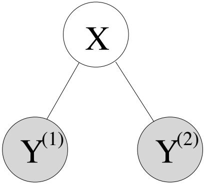


Figure: Modeling ambiguities (Navaratnam et al. 07)

Shared Kernel Information Embedding

- Extension of the Kernel Information Embedded (Memisevic 06) to have a shared latent space.
- The model is learn by maximizing the mutual information of a shared latent space $\mathbf{x}^{(i)}$ and an observation space $\mathbf{y}^{(i)}$



- The mutual information is approximated using kernel density estimation (KDE) as

$$\begin{aligned}\hat{I}\left(\mathbf{y}^{(i)},\mathbf{x}\right) &= -\frac{1}{N}\sum_j \log \sum_t k_x(\mathbf{x}_j,\mathbf{x}_t) - \frac{1}{N}\sum_j \log \sum_t k_y(\mathbf{y}_j^{(i)},\mathbf{y}_t^{(i)}) \\ &+ \frac{1}{N}\sum_j \log \sum_t k_x(\mathbf{x}_j,\mathbf{x}_t) k_y(\mathbf{y}_j^{(i)},\mathbf{y}_t^{(i)}) \ .\end{aligned}$$

Shared Kernel Information Embedding

- In the shared KIE model the loss function is defined as (Sigal et al. 09).

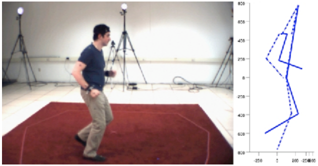
$$L_{data} = -\sum_{i=1}^{V} \hat{I}\left(\mathbf{y}^{(i)}, \mathbf{x}\right)$$

- For inference, the mean prediction from a joint latent coordinate to a view is given by

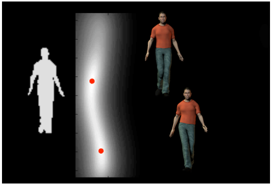
$$\bar{\mathbf{y}}_*^{(i)} = \sum_{j=1}^{N} \frac{k_x(\mathbf{x}_*, \mathbf{x}_j)}{\sum_{t=1}^{N} k_x(\mathbf{x}_*, \mathbf{x}_t)} \mathbf{y}_j^{(i)}$$

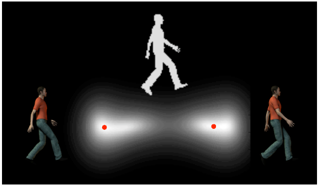
Human Pose Estimation

- We seek to recover the 3D pose from image features.



- The mapping is multimodal: an image observation can correspond to more than one pose.

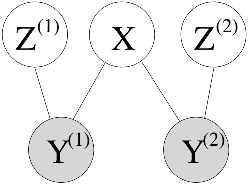




- Private latent spaces can model these ambiguities.

Shared and private information

- Ek et al. 08 developed NCCA
- First compute the shared space using CCA
- Then solve for the private space iteratively by solving an eigenvalue problem to reconstruct the residual information.



Shared and private information

- Use NCCA to initialize a GPLVM with shared and private spaces
- Problem, learning the GPLVM tends to merge information between shared and private

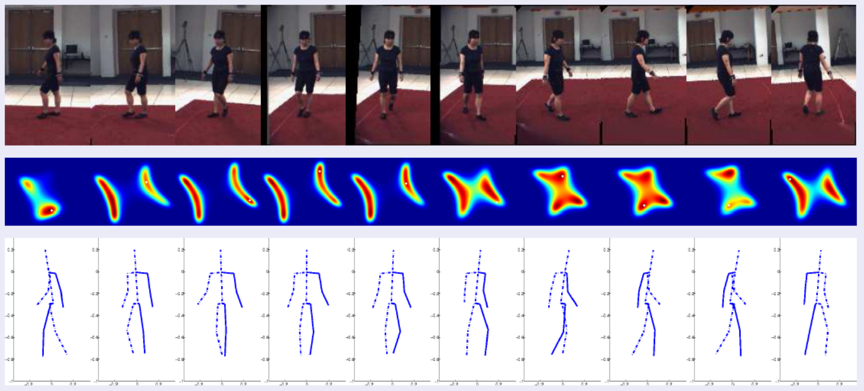
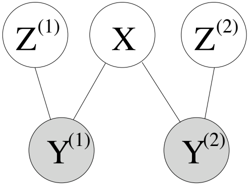


Figure: Modeling ambiguities (Ek et al. 08)

Factorized Orthogonal Latent Spaces (FOLS)

- Learn shared and private spaces that represent non-redundant information by means of orthogonality constraints (Salzmann et al. 10)
- Discover the structure and dimensionality of latent spaces by encourage low-dimensionality (Geiger et al. 09).
- Salzmann et al. demonstrate the effectiveness of our constraints on 2 different models: Shared GPLVM and Shared KIE.



- A FOLS model can be learned by minimizing

$$\mathcal{L} = L_{data} + L_{ortho} + L_{dim} + L_{energy}$$

Orthogonality

- We encourage the different latent spaces to be non-redundant.

$$L_{ortho} = \alpha \sum_{i} \left(||\mathbf{X}^{T} \cdot \mathbf{Z}^{(i)}||_F^2 + \sum_{j>i} ||(\mathbf{Z}^{(i)})^{T} \cdot \mathbf{Z}^{(j)}||_F^2 \right) .$$

- Minimize the Frobenius norm of inner product of latent spaces.
- This has the advantage of being continuous and differentiable.

Low-Dimensionality

- Encourage $\mathbf{M}^{(i)}$ to be low rank, with $\mathbf{m}^{(i)} = [\mathbf{x}, \mathbf{z}^{(i)}]$.
- Functions of the singular values s_i are typically used as relaxations.

$$L_{dim} = \gamma \sum_i \phi(s_i) \ .$$

- A particular instance of this is the trace norm, which is convex

$$\phi(s_i) = \sum_j |s_{i,j}| \ .$$

- L_{data} is non-convex, so we can use non-convex regularizers.

$$\phi(s_i) = \sum_j (1 + \beta \log(s_{i,j}^2)) \ .$$

- This drives smaller singular values faster to 0.

Energy Conservation

- Orthogonality and low-dimensionality terms tend to drive the latent coordinates to 0.
- We seek to conserve the energy of the observed data.

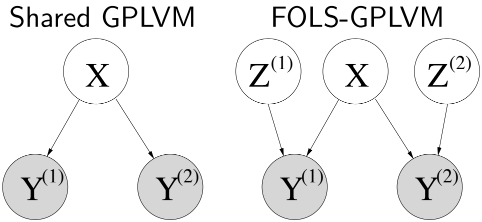
$$L_{energy} = \eta \sum_i (E_0^{(i)} - \sum_j s_{i,j}^2)^2 \ ,$$

where $E_0^{(i)} = \sum_j p_{i,j}^2$, with $p_{i,j}$ the singular values of $\mathbf{Y}^{(i)}$.

Data Term

- The data term L_{data} depends on the particular model into which we incorporate our constraints.
- Salzmann et al. 10 used two different models:
  - Shared Gaussian Process Latent Variable Model.
  - Shared Kernel Information Embedding.

FOLS-GPLVM

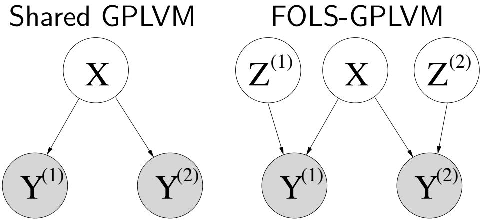


- We model the mapping from a joint latent space to an observation spaces as

$$p(\mathbf{Y}^{(i)}|\mathbf{Z}^{(i)}, \mathbf{X}) = \prod_{d=1}^{D_i} \mathcal{N}(\mathbf{Y}^{(i)}_{:,d}|0, \mathbf{K}^{(i)}) ,$$

  where $\mathbf{K}^{(i)}$ is an $N \times N$ kernel matrix.
- In practice we used the sum of an RBF kernel and a bias.

FOLS-GPLVM



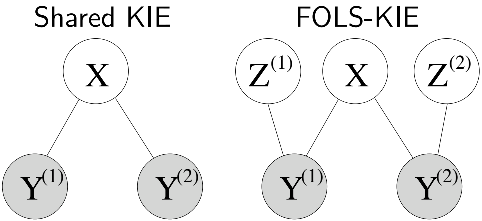
- In the FOLS-GPLVM, the loss function is defined as

$$L_{data} = \sum_{i=1}^{V} \left(\frac{D_i}{2} \ln |\mathbf{K}^{(i)}| + \frac{D_i}{2} tr \left[(\mathbf{K}^{(i)})^{-1} \mathbf{Y}^{(i)} (\mathbf{Y}^{(i)})^T \right] \right) \, .$$

- For inference, the mean prediction from a joint latent coordinate to a view is given by

$$\bar{\mathbf{y}}_*^{(i)} = (\mathbf{k}_*^{(i)})^T (\mathbf{K}^{(i)})^{-1} \mathbf{Y}^{(i)} \, .$$

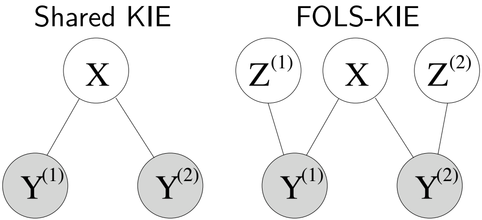
FOLS-KIE



- We seek to maximize the mutual information of a joint latent space $\mathbf{m}^{(i)}$ and an observation space $\mathbf{y}^{(i)}$.
- The mutual information is approximated using kernel density estimation (KDE) as

$$\begin{aligned}\hat{I}\left(\mathbf{y}^{(i)}, (\mathbf{x}, \mathbf{z}^{(i)})\right) &= -\frac{1}{N}\sum_j \log \sum_t k_m(\mathbf{m}_j^{(i)}, \mathbf{m}_t^{(i)}) - \frac{1}{N}\sum_j \log \sum_t k_y(\mathbf{y}_j^{(i)}, \mathbf{y}_t^{(i)}) \\ &+ \frac{1}{N}\sum_j \log \sum_t k_m(\mathbf{m}_j^{(i)}, \mathbf{m}_t^{(i)}) k_y(\mathbf{y}_j^{(i)}, \mathbf{y}_t^{(i)}) \;.\end{aligned}$$

FOLS-KIE



- In the FOLS-KIE, the loss function is defined as

$$L_{data} = -\sum_{i=1}^{V} \hat{l}\left(\mathbf{y}^{(i)}, (\mathbf{x}, \mathbf{z}^{(i)})\right) \quad .$$

- For inference, the mean prediction from a joint latent coordinate to a view is given by

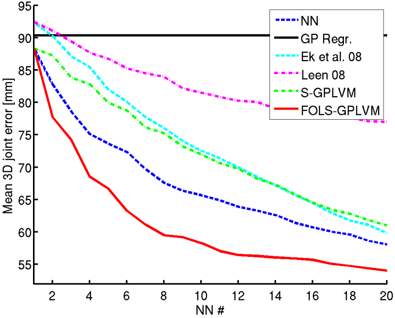
$$\bar{\mathbf{y}}_*^{(i)} = \sum_{j=1}^{N} \frac{k_m(\mathbf{m}_*{}^{(i)}, \mathbf{m}_j^{(i)})}{\sum_{t=1}^{N} k_m(\mathbf{m}_*{}^{(i)}, \mathbf{m}_t^{(i)})} \mathbf{y}_j^{(i)} \quad .$$

Human Pose Estimation

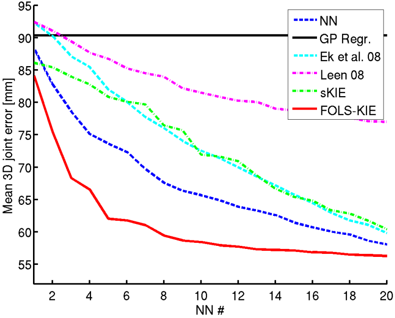
- Inference strategy
  - Find nearest neighbor in image features space.
  - Compute k-NN in shared space.
  - Take the corresponding private coordinates.
  - Infer the pose from the FOLS-GPLVM or FOLS-KIE equations.
- Baselines
  - k-NN in image features space.
  - GP regression.
  - Shared GPLVM or Shared KIE.
  - Shared-Private factorization (Ek et al. 2008, Leen 2008).

Human Pose Estimation

- Inference strategy
  - Find nearest neighbor in image features space.
  - Compute k-NN in shared space.
  - Take the corresponding private coordinates.
  - Infer the pose from the FOLS-GPLVM or FOLS-KIE equations.
- Baselines
  - k-NN in image features space.
  - GP regression.
  - Shared GPLVM or Shared KIE.
  - Shared-Private factorization (Ek et al. 2008, Leen 2008).

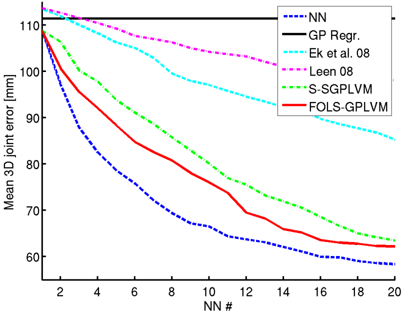


FOLS-GPLVM

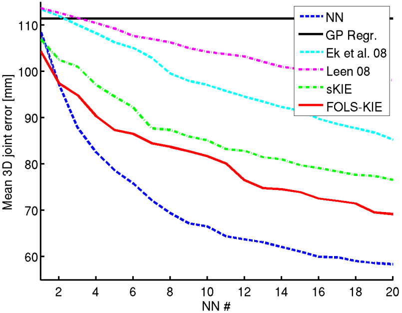


FOLS-KIE

Figure: Humaneva jog motion (Salzmann et al. 10)



FOLS-GPLVM



FOLS-KIE

Figure: Humaneva walk motion (Salzmann et al. 10)

Discriminative prediction

We have already covered

- NN
- Regression
- Mixture of experts
- Subspace models

Now we are going to see **structure prediction**

Non parametric BP for hand tracking

- We will focus on Sudderth et al. 04.
- Similar ideas for whole body in Sigal et al. 03.
- Accurately locating a few fingers highly constrains the set of possible global poses.
- GOAL: Robustly propagate local image evidence to track arbitrary hand motions.
- Use structure prediction and graphical models to solve this.

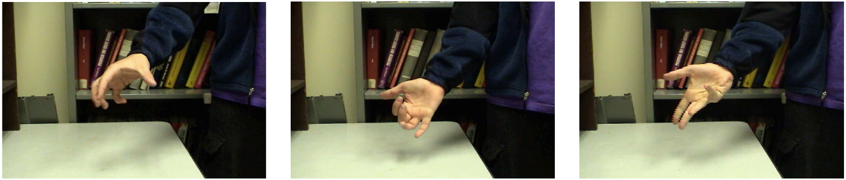


Figure: Sudderth et al. 04

Graphical models

An undirected graph $\mathcal{G}$ is defined by

- $\mathcal{V}$ the set of nodes $\{1, 2, \cdots, N\}$
- $\mathcal{E}$ the set of edges (i, j) connecting nodes $i, j \in \mathcal{V}$
- Nodes $i \in \mathcal{V}$ are associated with random variables $\mathbf{x}_i$
- Graph separation represents conditional independence

$$p(\mathbf{x}_A, \mathbf{x}_C | \mathbf{x}_B) = p(\mathbf{x}_A | \mathbf{x}_B) p(\mathbf{x}_C | \mathbf{x}_B)$$

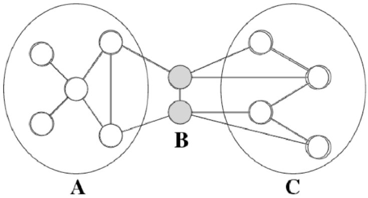


Figure: Sudderth 10

Pairwise Markov Random Fields

- Product of arbitrary positive clique potential functions
- Guaranteed Markov with respect to corresponding graph

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(\mathbf{x}_i, \mathbf{x}_j) \prod_{i \in \mathcal{V}} (\mathbf{x}_i, \mathbf{y})$$

- One case that we have seen in class is an HMM, where the dependency is temporal.

Belief Propagation (BP)

- **Beliefs**: Approximate posterior marginal distributions (product update)

$$\hat{p}(\mathbf{x}_i|\mathbf{y}) = \alpha\psi_i(\mathbf{x}_i, \mathbf{y}) \prod_{k\in\Gamma(i)} m_{ki}(\mathbf{x}_i)$$

  with $\Gamma(i)$ the neighborhood of node i.

- **Messages**: Approximate sufficient statistics (integral update)

$$m_{ij} = \alpha \int_{\mathbf{x}_i} \psi_{ji}(\mathbf{x}_j, \mathbf{x}_i)\psi(\mathbf{x}_i, \mathbf{y}) \prod_{k\in\Gamma(i)\setminus j} m_{ki}(\mathbf{x}_i) d\mathbf{x}_i = \alpha \int_{\mathbf{x}_i} \psi_{ji}(\mathbf{x}_j, \mathbf{x}_i) \frac{\hat{p}(\mathbf{x}_i|\mathbf{y})}{m_{ji}(\mathbf{x}_i)} d\mathbf{x}_i$$

- BP is exact for trees.

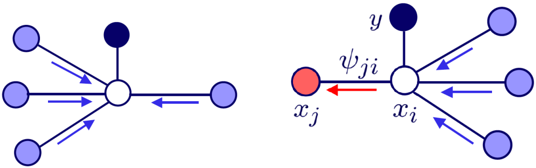


Figure: Sudderth 10

Messages for continuous variables

$$m_{ij} = \alpha \int_{\mathbf{x}_i} \psi_{ji}(\mathbf{x}_j, \mathbf{x}_i) \psi(\mathbf{x}_i, \mathbf{y}) \prod_{k \in \Gamma(i) \setminus j} m_{ki}(\mathbf{x}_i) d\mathbf{x}_i$$

Discrete State Variables

- Messages are finite vectors
- Updated via matrix-vector products

Gaussian State Variables

- Messages are mean and covariance
- Updated via information Kalman filter

Continuous Non-Gaussian State Variables

- Closed parametric forms unavailable
- Discretization can be intractable even with 2 or 3 dimensional states

Messages for continuous variables

- Discrete State Variables
- Gaussian State Variables
- Continuous Non-Gaussian State Variables

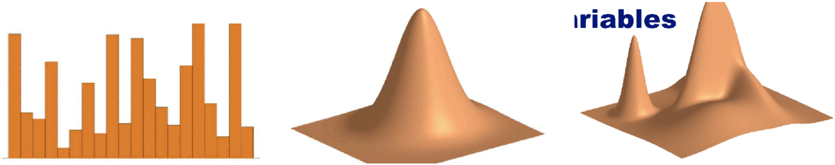


Figure: Message representation as (left) discrete (center) Gaussian and (right) continuous non-Gaussian state variables (Sudderth 10)

Non-parametric Inference for General Graphs

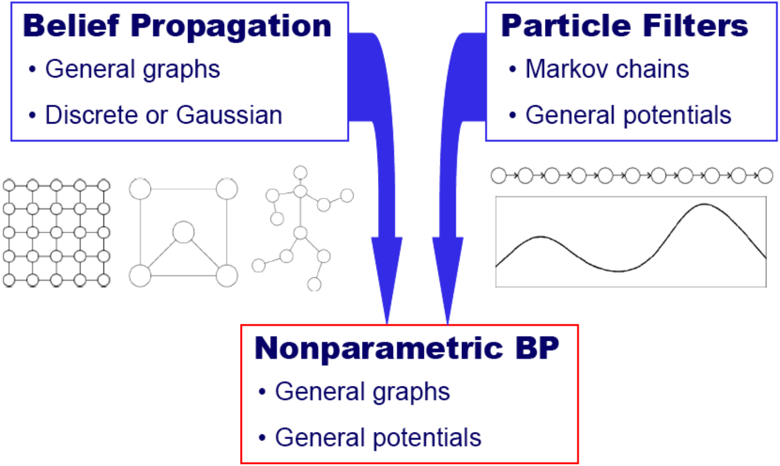


Figure: Non-parametric Inference for General Graphs (Sudderth 10)

Nonparametric Density Estimates

- Kernel (Parzen Window) Density Estimator approximates PDF by a set of smoothed data samples

$$\hat{p}(x) = \frac{1}{M} \sum_{i=1}^{M} \frac{1}{\sigma} K\left(\frac{x - X_i}{\sigma}\right)$$

where X_i are M independent samples from $p(x)$, K is a kernel, typically Gaussian, and σ is the bandwidth

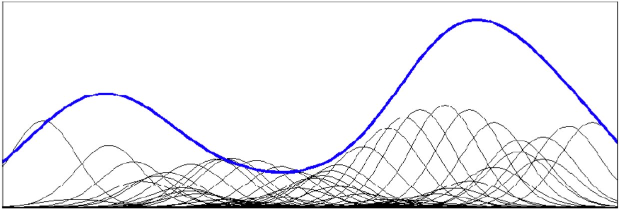


Figure: Kernel density estimation (Sudderth 10)

Nonparametric BP

- Input messages are kernel density estimates (Gaussian)
- Message product: draw L samples

$$\mathbf{x}_i^{(l)} \sim \psi_i(\mathbf{x}_i, \mathbf{y}) \prod_{k \in \Gamma(i) \setminus j} m_{ki}(\mathbf{x}_i)$$

- Message propagation: Monte Carlo integration

$$\mathbf{x}_j^{(l)} \sim \psi_{ji}(\mathbf{x}_j, \mathbf{x}_i^{(l)})$$

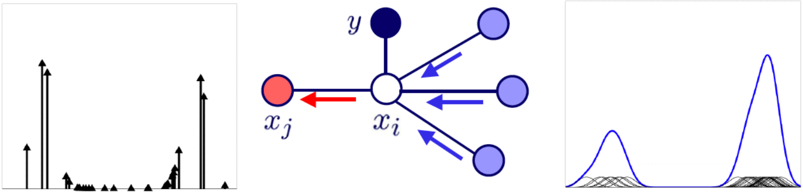


Figure: Non Parametric BP (Sudderth 10)

Nonparametric BP

- Output message estimated from weighted samples via a bandwidth selection rule

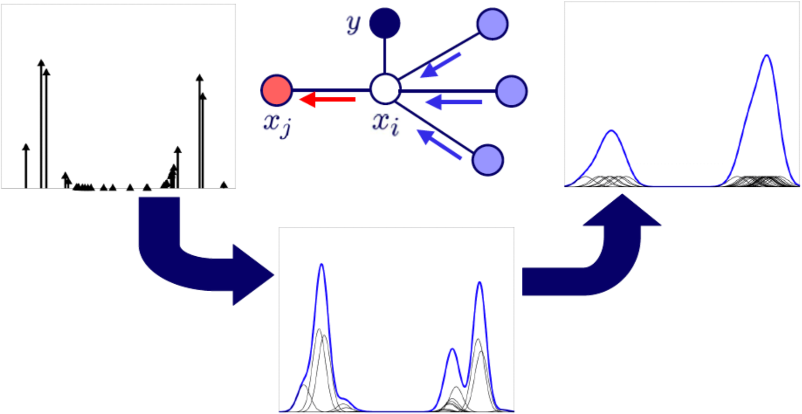


Figure: Non Parametric BP (Sudderth 10)

NBP Marginal Update

Importance Sampling

- Sample from product of all Gaussian mixture messages
- Reweight samples by likelihoods (like particle filter)

$$\mathbf{x}_i^{(l)} \sim \psi_i(\mathbf{x}_i, \mathbf{y}) \prod_{k \in \Gamma(i)} m_{ki}(\mathbf{x}_i)$$

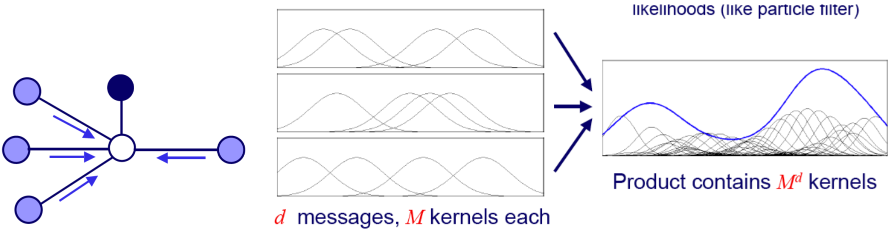


Figure: NBP Marginal Update (Sudderth 10)

Structural model

- Hand described by 16 rigid bodies
- 3D geometry of each rigid body modeled by truncated quadric surfaces: Ellipsoids, cones and cylinders (Stenger et al. 01).
- Perspective projection maps quadrics to conics (ellipses, pairs of lines, etc.) for efficient computation of edge and silhouettes.
- Fixed geometry measured offline

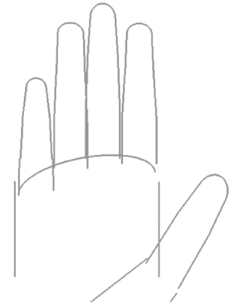


Figure: Sudderth et al. 04

Hand model projections

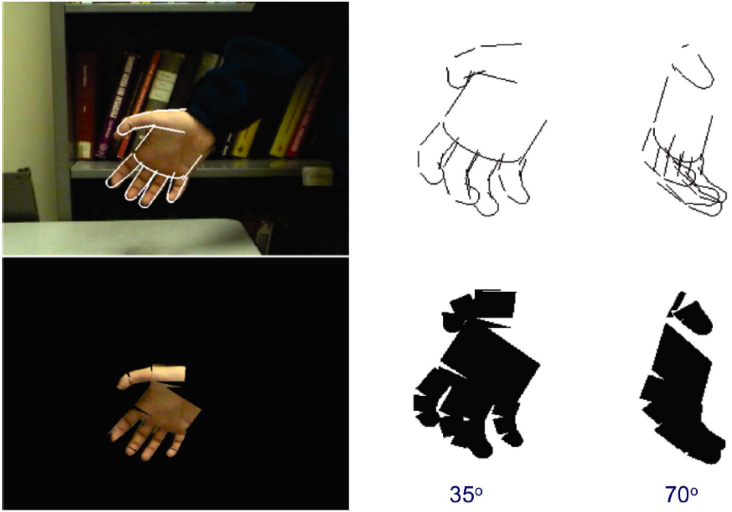


Figure: Hand model projections (Sudderth et al. 04)

Graphical model

- We create the graphical model from constraints

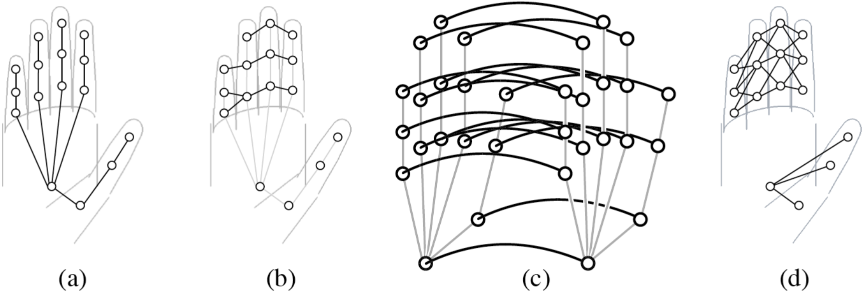


Figure: Hand constraints, (a) kinematic, (b) structural, (c) dynamic and (d) occlusion (Sudderth et al. 04)

Kinematic model

- Rigid bodies kinematically related by revolute joints
- Model has total of 26 DOF: 20 joint angles (4 per finger), Palms global position and orientation.
- Likelihood calculation requires global coordinates of all bodies: No direct evidence for joint angle.
- Forward kinematics maps joint angles to 3D poses.
- The nodes are rigid bodies and the edges joints

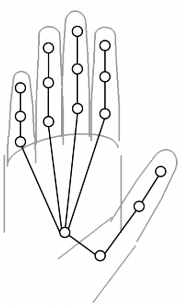


Figure: Sudderth et al. 04

Local State Representation

- The hand has 16 joints $\mathbf{x} = \{\mathbf{x}_1, \cdots, \mathbf{x}_{16}\}$.
- Each joint is described with a redundant parameterization $\mathbf{x}_i = [\mathbf{q}_i, \mathbf{r}_i]$
- $\mathbf{q}_i$ is a 3D position, and $\mathbf{r}_i$ is a quaternion.
- Advantage: Image appearance directly relates to local state
- Disadvantage: It's redundant, we have additional dof.

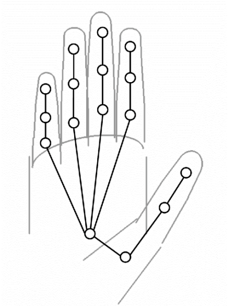


Figure: Sudderth et al. 04

Kinematic Constraints

- Define an indicator function for each joint edge $(i, j) \in \mathcal{E}_K$

$$\psi^K_{i,j}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \text{ valid} \\ 0 & \text{otherwise} \end{cases}$$

- Kinematic prior model:

$$p_K(\mathbf{x}) = \prod_{(i,j)\in\mathcal{E}_K} \psi^K_{i,j}(\mathbf{x}_i, \mathbf{x}_j)$$

- Graphical model exactly enforcing original joint angle constraints, e.g., conditioned on the palm, the fingers are statistically independent

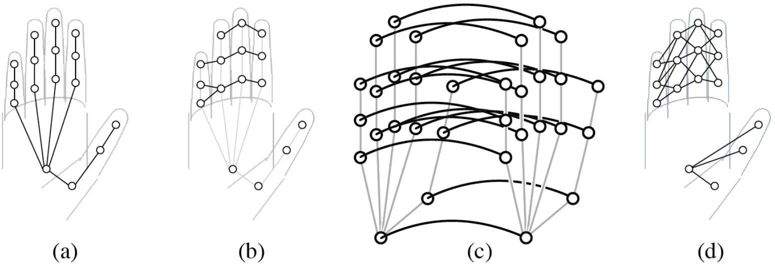


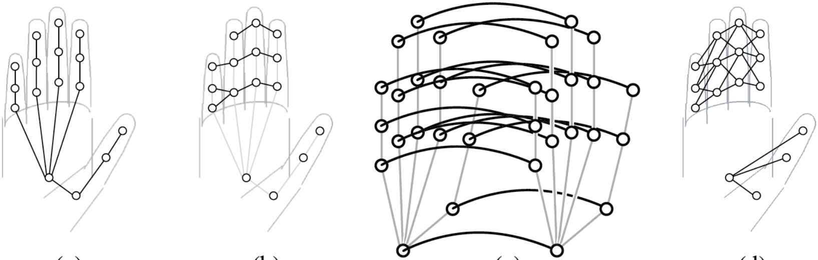
Figure: Sudderth et al. 04

Structural Constraints

- Kinematics do not prevent finger intersection (joints not independent)
- Ideal structural constraint prevents 3D quadric surface intersection

$$\psi^S_{i,j}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 & \text{if } ||\mathbf{q}_i - \mathbf{q}_j|| > \delta_{i,j} \\ 0 & \text{otherwise} \end{cases}$$

- Structural prior model: $$p_S(\mathbf{x}) = \prod_{(i,j)\in\mathcal{E}_S} \psi^S_{i,j}(\mathbf{x}_i, \mathbf{x}_j)$$



(a) (b) (c) (d)

Figure: Sudderth et al. 04

Observation model

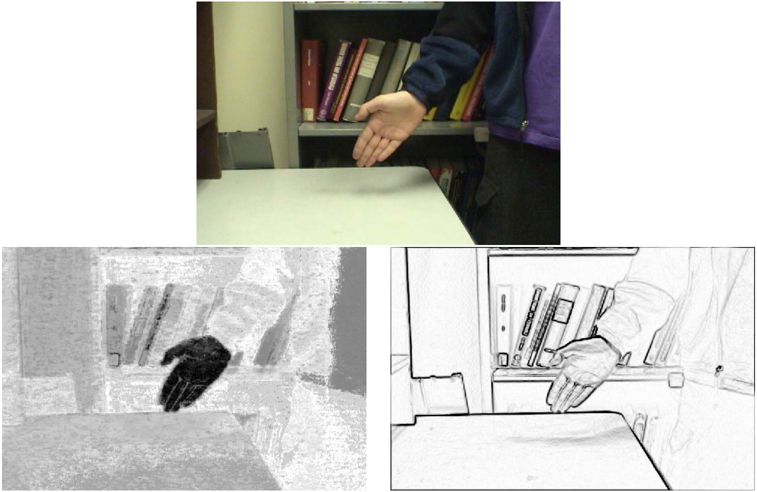


Figure: Observation model, (a) original image, (b) skin color, (c) edge intensity (Sudderth et al. 04)

Silhouette Matching: Skin Color

- Assume RGB values at each pixel independent
- p_{skin} is the histogram estimated from labeled skin pixels
- p_{bkgd} is the histogram estimated from hand-free background images

$$p_C(\mathbf{y}|\mathbf{x}) = \prod_{u \in \Omega(\mathbf{x})} p_{skin}(u) \prod_{v \in \Upsilon \setminus \Omega(\mathbf{x})} p_{bkgd}(v) \propto \prod_{u \in \Omega(\mathbf{x})} \frac{p_{skin}(u)}{p_{bkgd}(u)}$$

where $\Omega(\mathbf{x})$ are the pixels in the silhouette projected from $\mathbf{x}$, and Υ is the set of all pixels.

- Only evaluate likelihood ratio over projected silhouette

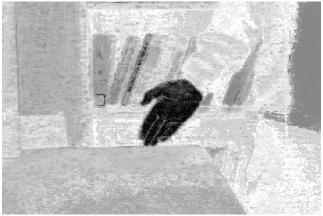


Figure: Sudderth et al. 04

Edge Matching: Steered Gradient

- Steer derivative of Gaussian response to orientation of projected hand boundary.
- p_{edge} is the histogram estimated from labeled edge pixels.
- p_{bkgd} is the histogram estimated from background images.

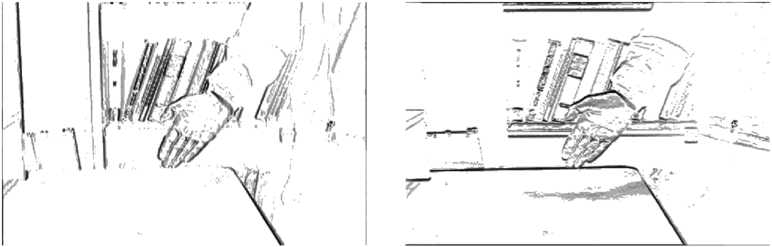


Figure: Derivatives with respect to the horizontal and vertical axis (Sudderth et al. 04)

Local Likelihood Decomposition

- If two hand components do not occlude each other, they will project to disjoint subsets of the image

$$p_C(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{16} p_C(\mathbf{y}|\mathbf{x}_i) \propto \prod_{u \in \Omega(\mathbf{x})} \frac{p_{skin}(u)}{p_{bkgd}(u)} = \prod_{i=1}^{16} \prod_{u \in \Omega(\mathbf{x}_i)} \frac{p_{skin}(u)}{p_{bkgd}(u)}$$

- Edge likelihood ratio decomposes similarly
- Reasoning about self-occlusions discussed later ...

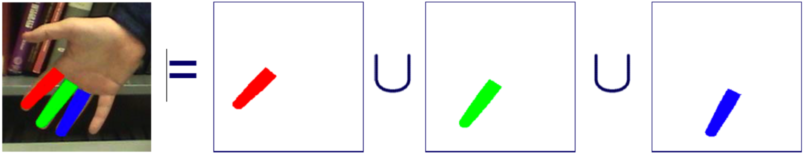
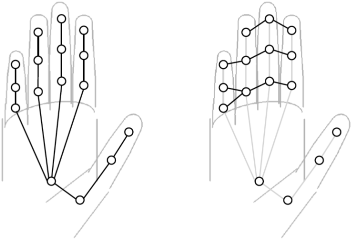


Figure: Sudderth 10

Inferring Hand Position

- When using kinematic and structural constraints the posterior can be computed as

$$p(\mathbf{x}|\mathbf{y}) \propto p_K(\mathbf{x}) p_S(\mathbf{x}) \left[\prod_{i=1}^{16} \underbrace{p_C(\mathbf{y}|\mathbf{x}_i) p_E(\mathbf{y}|\mathbf{x}_i)}_{\text{Color and edge}} \right]$$



- Pairwise Markov Random Field

$$p(\mathbf{x}|\mathbf{y}) = \frac{1}{Z} \prod_{(i,j)\in\mathcal{E}} \psi_{i,j}(\mathbf{x}_i, \mathbf{x}_j) \prod_{i\in\mathcal{V}} \psi_i(\mathbf{x}_i, \mathbf{y})$$

NBP Hand Tracker Marginal Update

Importance Sampling

- Sample from product of all Gaussian mixtures
- Reweight samples by analytic functions (like particle filter)

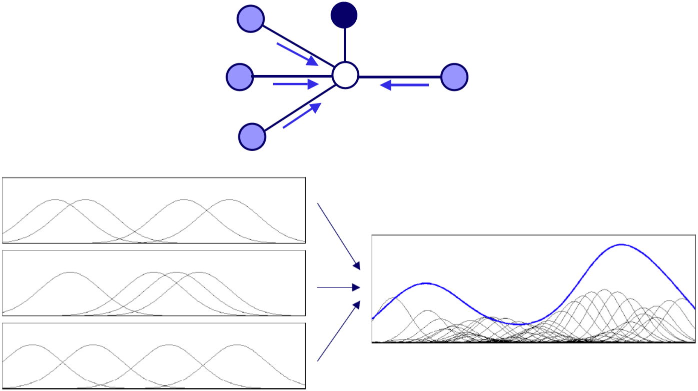


Figure: Sudderth 10

- Start with weighted samples $\mathbf{x}_i^{(l)}$ from last marginal update
- Kinematic potential gives all valid poses equal weight
- Sample uniformly among allowable joint angles θ.
- Compute corresponding pose of $\mathbf{x}_j$ by forward kinematics

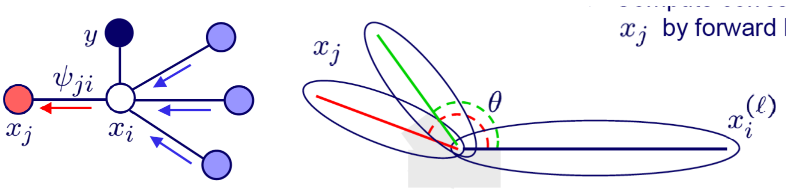


Figure: Kinematic message propagation (Sudderth 10)

Structural Message Propagation

- Exact: Integrate belief over all poses outside some ball centered at the candidate pose $\mathbf{x}_j$
- Approximate: Sum weights of all Gaussians with centers outside that ball

$$m_{ij}(\mathbf{x}_j) = \alpha \int_{\mathbf{x}_i} \psi^S_{j,i}(\mathbf{x}_j, \mathbf{x}_i) \frac{\hat{p}(\mathbf{x}_i|\mathbf{y})}{m_{ji}(\mathbf{x}_i)} d\mathbf{x}_i$$

- Reduces weight of particles which overlap with likely positions of neighboring nodes

$$\psi^S_{i,j}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 & \text{if } ||\mathbf{q}_i - \mathbf{q}_j|| > \delta_{i,j} \\ 0 & \text{otherwise} \end{cases}$$

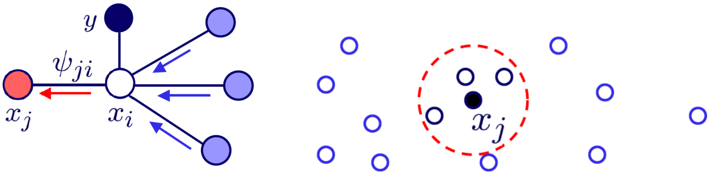


Figure: Structural message propagation (Sudderth 10)

Single Frame Inference

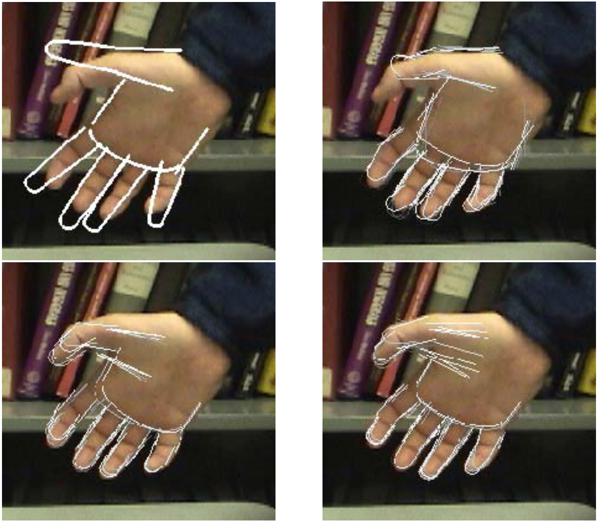


Figure: Single frame estimation (Sudderth et al. 04)

Self Occlusion Mask

- Condition on occlusion mask **z** allows exact likelihood decomposition

$$p_C(\mathbf{y}|\mathbf{x}) \propto \prod_{i=1}^{16} \prod_{u \in \Omega(\mathbf{x}_i)} \left(\frac{p_{skin}(u)}{p_{bkgd}(u)} \right)^{z_{i(u)}}$$

where the occlusion variables

$$z_{i(u)} = \begin{cases} 1 & \text{if pixel } u \text{ in the projection of body i is occluded} \\ 0 & \text{otherwise} \end{cases}$$

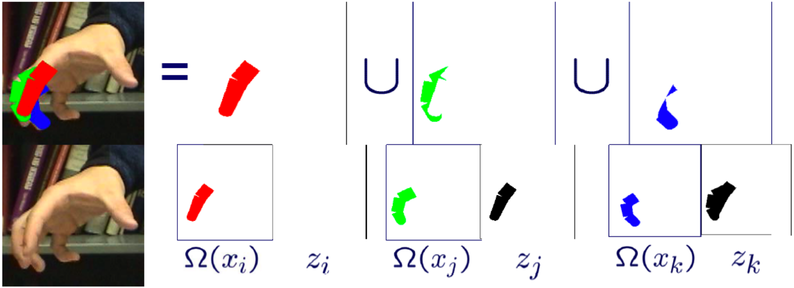


Figure: Sudderth 10

Distributed Occlusion Reasoning

- Factor graph imposes constraints ensuring occlusion consistency
- Use BP to analytically estimate probability of pixels occlusion

$$v_{i(u)} = Pr[z_{i(u)} = 0]$$

- Neglecting correlations among the occlusion variables, the likelihood function (integrating over occlusions) becomes

$$p_C(\mathbf{y}|\mathbf{x}_i) \propto \prod_{u \in \Omega(\mathbf{x}_i)} \left[v_{i(u)} \underbrace{(1)}_{uninformative} + (1 - v_{i(u)}) \underbrace{\left(\frac{p_{skin}(u)}{p_{bkgd}(u)} \right)}_{skincolor} \right]$$

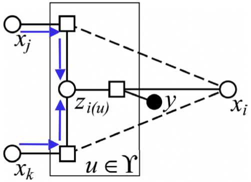


Figure: Sudderth 10

Occlusion Reasoning Example

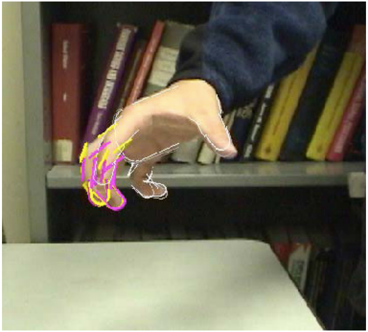
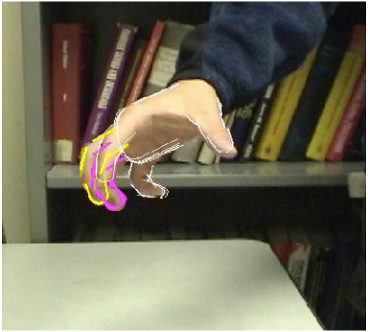
 

Figure: Pose estimation (left) without and (right) with occlusion reasoning. The middle finger is depicted in yellow and the Ring finger in pink (Sudderth et al. 04)

Temporal Constraints and Tracking

- Add Gaussian potentials between adjacent time steps

$$\psi(\mathbf{x}_{t-1,i}, \mathbf{x}_{t,i}) = \mathcal{N}(\mathbf{x}_{t-1,i}|0, \mathbf{A}_{t,i})$$

- This can be interpreted as maximum entropy model given marginal variances in 3D pose ...
- ... or random walks implicitly coupled by kinematic and structural constraints

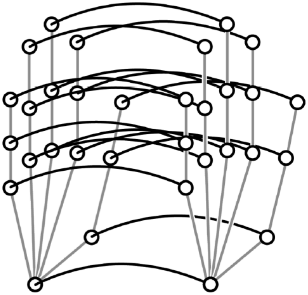


Figure: Temporal constraints (Sudderth et al. 04)

Tracking Hand Rotation (Sudderth et al 04)

Tracking Finger Motion (Sudderth et al 04)

Conclusions on structure prediction with NBP

Nonparametric Belief Propagation

- Inference in continous, non-Gaussian graphical models
- Very flexible, easy to adapt to diverse applications
- Multiscale samplers lead to computational efficiency

Framework for Tracking Problems

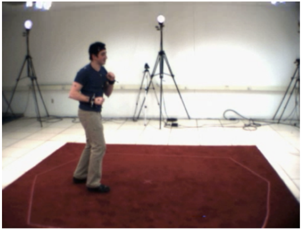
- Modular state representation
- Graphical model of kinematics, structure, and dynamics
- NBP may accommodate complexities such as occlusions
- Many other potential applications

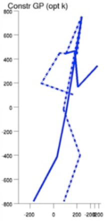
Code available online $http://ssg.mit.edu/nbp/$

MRF with discretization

- Use discrete MRF to choose within a set of poses

Articulated pose estimation





Discriminative Approaches

- \+ Allow for any image representation
- – Require large training sets
- – Assume output dimensions are independent given the inputs

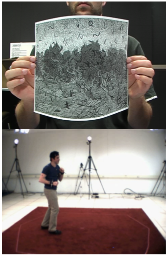
Generative Approaches

- \+ Yield better accuracy
- – Require good initialization

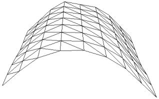
Combining Generative and Discriminative

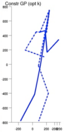
- Discriminative and generative methods should be used together.
- This was observed in the past, however
  - [Sminchisescu et al. 06] rely on the generative only for training,
  - [Rosales et al. 06] and [Sigal et al. 07] rely on the discriminative only for initialization.
- We would like a more principled combination of generative and discriminative methods.

Our Approach



1) Discriminative

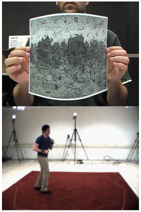




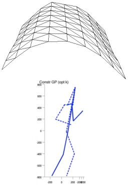
3) Generative

2) Constraints

Discriminative Regression



$\xrightarrow{\mathbf{f}}$

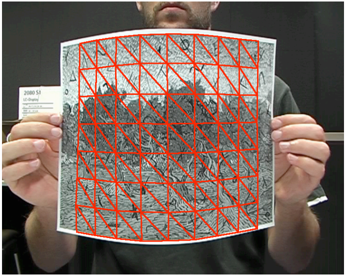


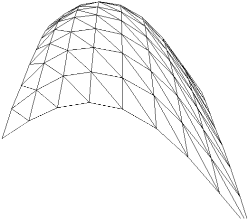
$\mathbf{x}$ $\mathbf{y}$

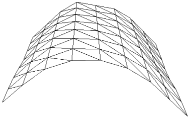
- Discriminative methods focus on learning an estimate $\hat{\mathbf{f}}$ of the mapping $\mathbf{y} = \mathbf{f}(\mathbf{x}) + \epsilon$ from training data.
- Given a new input $\mathbf{x}_*$, $\mathbf{y}$ is computed as the prediction $\hat{\mathbf{f}}(\mathbf{x}_*)$.
- When $\mathbf{y}$ is multi-dimensional, the outputs are typically assumed to be independent.

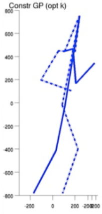
Discriminative Regression: Limitations

- The outputs independence assumption yields estimations that do not satisfy some known constraints.









$\mathbf{y}$

- We seek to improve the discriminative prediction by introducing explicit constraints.
- In particular, we enforce the distances between pairs of 3D points $(\mathbf{y}_j, \mathbf{y}_k)$ to remain constant.

$$\min_{\mathbf{y}} \ ||\hat{\mathbf{f}}(\mathbf{x}_*) - \mathbf{y}||_2^2$$

$$\text{subject to} \ ||\mathbf{y}_k - \mathbf{y}_j||_2^2 = l_{j,k}^2 \ , \ \forall (j,k) \in \mathcal{E} \ ,$$

where $\mathcal{E}$ is the set of constrained link and $l_{j,k}$ are the known distances.

Constrained Discriminative Regression

- Our optimization problem is non-convex due the constraints:

$$c_{jk}(\mathbf{y}) = ||\mathbf{y}_k - \mathbf{y}_j||_2^2 = l_{j,k}^2 \ , \ \forall (j,k) \in \mathcal{E}$$

- We iteratively approximate the constraints $c_{jk}(\mathbf{y})$ with their first order Taylor expansion

$$c_{jk}(\mathbf{y}_{t+1}) = c_{jk}(\mathbf{y}_t) + \nabla c_{jk}(\mathbf{y}_t)\delta\mathbf{y}_t = l_{j,k}^2 \ .$$

- At each iteration t, we compute the constraints Jacobian matrix $\mathbf{J}_t$ and the constraint errors $\mathbf{g}_t$, and seek a displacement $\delta\mathbf{y}_t$, such that

$$\mathbf{J}_t\delta\mathbf{y}_t = \mathbf{g}_t \ .$$

Constrained Discriminative Regression

- The previous system has more unknowns than constraints.
- Therefore it defines the family of solutions

$$\mathbf{s}(\gamma_t) = \mathbf{y}_t + \mathbf{J}_t^+ \mathbf{g}_t + \mathbf{V}_t^T \gamma_t \ ,$$

where $\mathbf{J}_t^+$ is the pseudo-inverse of $\mathbf{J}_t$,and $\mathbf{V}_t$ contains the right singular vectors of $\mathbf{J}_t$ which have zero-valued singular values.

- Given the new unknowns γ_t that implicitly minimize the constraints violation, we re-write our problem as

$$\gamma_t^* = \underset{|}{\operatorname{argmin}} \ |\hat{\mathbf{f}}(\mathbf{x}_*) - \mathbf{s}(\gamma_t)||_2^2 \ ,$$

which has a closed-form solution.

Constrained Discriminative Regression

- The previous system has more unknowns than constraints.
- Therefore it defines the family of solutions

$$\mathbf{s}(\gamma_t) = \mathbf{y}_t + \mathbf{J}_t^{+}\mathbf{g}_t + \mathbf{V}_t^T \gamma_t \ ,$$

where $\mathbf{J}_t^{+}$ is the pseudo-inverse of $\mathbf{J}_t$,and $\mathbf{V}_t$ contains the right singular vectors of $\mathbf{J}_t$ which have zero-valued singular values.

- Given the new unknowns γ_t that implicitly minimize the constraints violation, we re-write our problem as

$$\gamma_t^* = \operatorname*{argmin}_{|} |\hat{\mathbf{f}}(\mathbf{x}_*) - \mathbf{s}(\gamma_t)||_2^2 \ ,$$

which has a closed-form solution.

Algorithm

$\mathbf{y}_1 = \hat{\mathbf{f}}(\mathbf{x}_*)$

for $t = 1$ to *iters* **do**

Compute the constraints Jacobian matrix $\mathbf{J}_t$

Compute the constraints errors $\mathbf{g}_t$

$\gamma_t = \mathrm{argmin}\, ||\hat{\mathbf{f}}(\mathbf{x}_*) - (\mathbf{y}_t + \mathbf{J}_t^+ \mathbf{g}_t + \mathbf{V}_t^T \gamma_t)||_2^2$

$\mathbf{y}_{t+1} = \mathbf{y}_t + \mathbf{J}_t^+ \mathbf{g}_t + \mathbf{V}_t^T \gamma_t$

end for

Better Use of the Predictor

- The approach described above depends on the predictor only through its fixed prediction $\hat{\mathbf{f}}(\mathbf{x}_*)$.
- We propose to rely on the *Representer theorem* which states that

$$\hat{f}(\mathbf{x}_*) = \sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \mathbf{x}_*) = \alpha \mathbf{k}_* \ ,$$

  where k is a kernel function and α is learned from the N training examples.
- For multi-dimensional outputs, we can write $\mathbf{y} = \hat{\mathbf{f}}(\mathbf{x}_*) = \boldsymbol{\alpha} \mathbf{k}_*$, with $\boldsymbol{\alpha} \in \Re^{D \times N}$.

Better Use of the Predictor

- We can rely more strongly on the learned predictor by treating $\mathbf{k}_*$ as an unknown.
- This lets us re-write our optimization problem as

$$\min_{\mathbf{k}_*} \ ||\hat{\mathbf{f}}(\mathbf{x}_*) - \alpha \mathbf{k}_*||_2^2$$
$$\text{subject to } \ ||\mathbf{y}_k(\mathbf{k}_*) - \mathbf{y}_j(\mathbf{k}_*)||_2^2 = l_{j,k}^2 \ , \ \forall (j,k) \in \mathcal{E} \ .$$

- Following a similar approach as before, we iteratively compute the Taylor expansion of our constraints with respect to $\mathbf{k}_*$.
- This yields a family of solutions characterized as

$$\mathbf{s}(\gamma_t) = \alpha \cdot \left(\mathbf{k}_{*,t} + \bar{\mathbf{J}}_t^+ \bar{\mathbf{g}}_t + \bar{\mathbf{V}}_t^T \gamma_t \right) \ .$$

- The optimal γ_t can still be obtained in closed-form.

Better Use of the Predictor

- We can rely more strongly on the learned predictor by treating $\mathbf{k}_*$ as an unknown.
- This lets us re-write our optimization problem as

$$\begin{aligned} \min_{\mathbf{k}_*} \quad & ||\hat{\mathbf{f}}(\mathbf{x}_*) - \alpha \mathbf{k}_*||_2^2 \\ \text{subject to} \quad & ||\mathbf{y}_k(\mathbf{k}_*) - \mathbf{y}_j(\mathbf{k}_*)||_2^2 = l_{j,k}^2 \ , \ \forall (j,k) \in \mathcal{E} \ . \end{aligned}$$

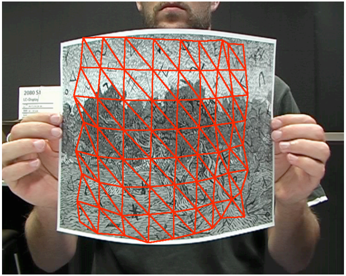
- Following a similar approach as before, we iteratively compute the Taylor expansion of our constraints with respect to $\mathbf{k}_*$.
- This yields a family of solutions characterized as

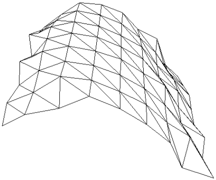
$$\mathbf{s}(\gamma_t) = \alpha \cdot \left(\mathbf{k}_{*,t} + \bar{\mathbf{J}}_t^+ \bar{\mathbf{g}}_t + \bar{\mathbf{V}}_t^T \gamma_t \right) \ .$$

- The optimal γ_t can still be obtained in closed-form.

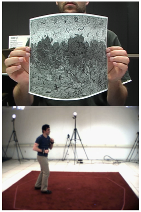
Poor Use of the Image

- One drawback of this method is that it only uses image information through the prediction of the discriminative method.
- The recovered pose will satisfy the constraints, but may have drifted away from the pose depicted in the image.





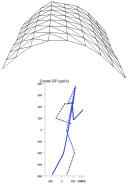
Combining Generative and Discriminative



$\mathbf{x}_*$

$\longleftarrow$

Constraints



$\mathbf{s}(\gamma_t)$

- At each iteration t, given the new variable γ_t, we solve

$$\min_{\gamma_t} \quad \mathcal{L}(\cdot, \gamma_t) + \lambda ||\hat{\mathbf{f}}(\mathbf{x}_*) - \mathbf{s}(\gamma_t)||_2^2 \ ,$$

where $\mathcal{L}(\cdot, \gamma_t)$ is an image-based loss function.

Image-based Loss Functions

In practice, we implemented 3 different image loss functions.

- Inverse mapping
  - Learn an estimate $\hat{\mathbf{h}}$ of the mapping $\mathbf{x} = \mathbf{h}(\mathbf{y}) + \epsilon$.
  - $\mathcal{L}(\mathbf{x}_*, \gamma_t) = ||\mathbf{x}_* - \hat{\mathbf{h}}(\mathbf{s}(\gamma_t))||_2^2$.
- 3D-2D correspondences
  - $\mathcal{L}(\gamma_t) = ||\mathbf{M}\mathbf{s}(\gamma_t) - \mathbf{b}||_2^2$.
  - Closed-form solution.

Image-based Loss Functions

In practice, we implemented 3 different image loss functions.

- Inverse mapping
  - Learn an estimate $\hat{\mathbf{h}}$ of the mapping $\mathbf{x} = \mathbf{h}(\mathbf{y}) + \epsilon$.
  - $\mathcal{L}(\mathbf{x}_*, \gamma_t) = ||\mathbf{x}_* - \hat{\mathbf{h}}(\mathbf{s}(\gamma_t))||_2^2$.
- 3D-2D correspondences
  - $\mathcal{L}(\gamma_t) = ||\mathbf{M}\mathbf{s}(\gamma_t) - \mathbf{b}||_2^2$.
  - Closed-form solution.
- More complete image representation
  - Template matching.
  - Edge information.

Image-based Loss Functions

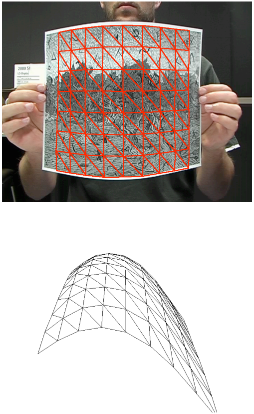
In practice, we implemented 3 different image loss functions.

- Inverse mapping
  - Learn an estimate $\hat{\mathbf{h}}$ of the mapping $\mathbf{x} = \mathbf{h}(\mathbf{y}) + \epsilon$.
  - $\mathcal{L}(\mathbf{x}_*, \gamma_t) = ||\mathbf{x}_* - \hat{\mathbf{h}}(\mathbf{s}(\gamma_t))||_2^2$.
- 3D-2D correspondences
  - $\mathcal{L}(\gamma_t) = ||\mathbf{Ms}(\gamma_t) - \mathbf{b}||_2^2$.
  - Closed-form solution.
- More complete image representation
  - Template matching.
  - Edge information.

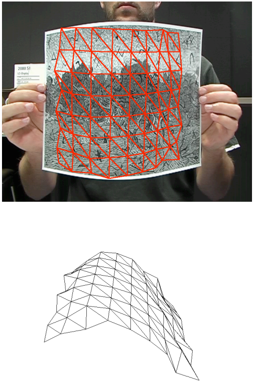
Algorithm

$\mathbf{y}_1 = \hat{\mathbf{f}}(\mathbf{x}_*)$, or $\mathbf{k}_{*,1} = \mathbf{k}_*$
for $t = 1$ to *iters* **do**
  Compute the constraints Jacobian matrix $\mathbf{J}_t$, or $\bar{\mathbf{J}}_t$
  Compute the constraints errors $\mathbf{g}_t$, or $\bar{\mathbf{g}}_t$
  $\gamma_t = \mathrm{argmin}\, \mathcal{L}(\cdot, \gamma_t) + \lambda ||\hat{\mathbf{f}}(\mathbf{x}_*) - \mathbf{s}(\gamma_t)||_2^2$
  Compute $\mathbf{y}_{t+1} = \mathbf{y}_t + \mathbf{J}_t^+ \mathbf{g}_t + \mathbf{V}_t^T \gamma_t$, or $\mathbf{k}_{*,t+1} = \mathbf{k}_{*,t} + \bar{\mathbf{J}}_t^+ \bar{\mathbf{g}}_t + \bar{\mathbf{V}}_t^T \gamma_t$
end for

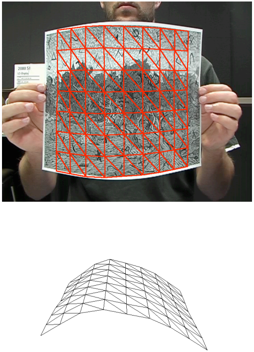
Comparison with Previous Reconstructions



Discriminative



Constrained Discr.



Constrained Discr. + Gen.

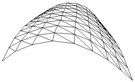
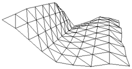
Our Choice of Predictor

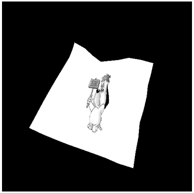
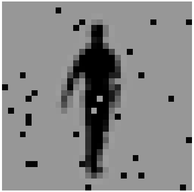
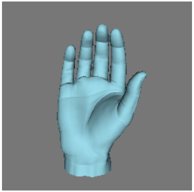
- In practice, we used Gaussian processes as our discriminative predictor.
- In this case, the basis $\boldsymbol{\alpha}$ can be computed in closed form as

$$\boldsymbol{\alpha} = \mathbf{Y}^T \mathbf{K}^{-1} ,$$

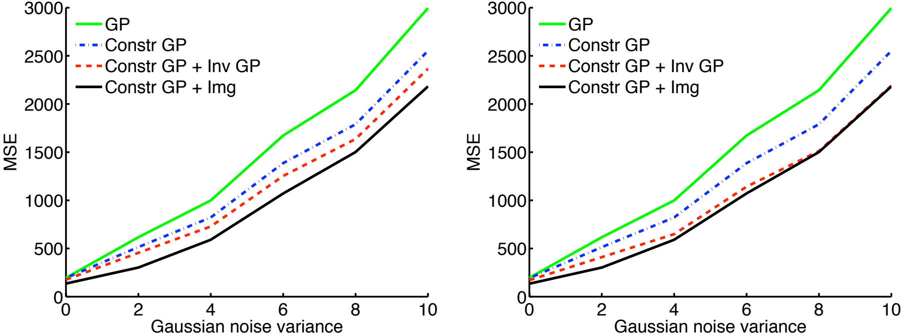
  where $\mathbf{Y} \in \Re^{N \times D}$ is the matrix of training outputs (e.g., poses), and $\mathbf{K}$ is the covariance matrix formed by evaluating the kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$ on the training inputs.
- Our kernel was taken to be the sum of an RBF kernel and a bias.

Experimental Evaluation

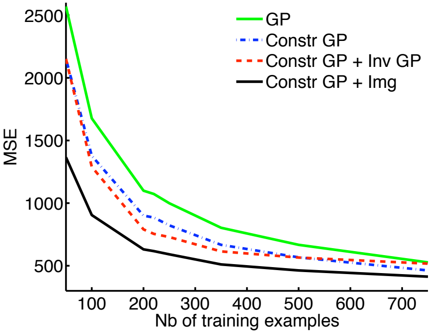





Reconstructing a Piece of Cardboard from 2D Locations



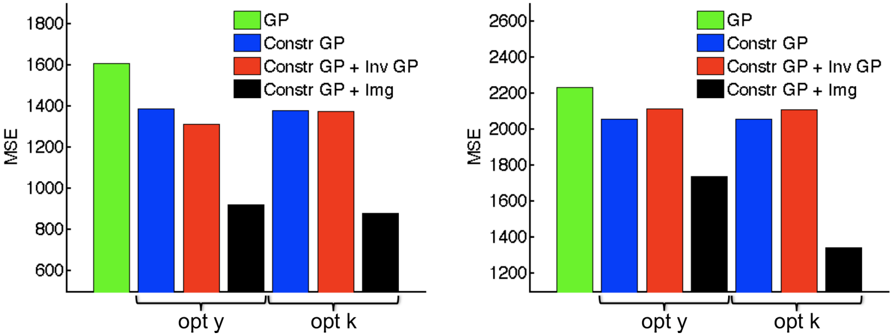
MSE as a function of the 2D noise variance when optimizing $\mathbf{y}$ (left), or $\mathbf{k}_*$ (right).

Reconstructing a Piece of Cardboard from 2D Locations



MSE as a function of the number of training examples when optimizing $\mathbf{y}$ (left), or $\mathbf{k}_*$ (right).

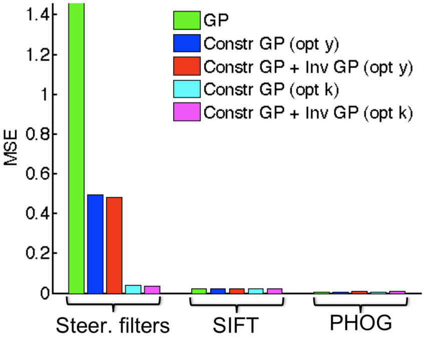
Non-Rigid Reconstruction from Pyramid HOG

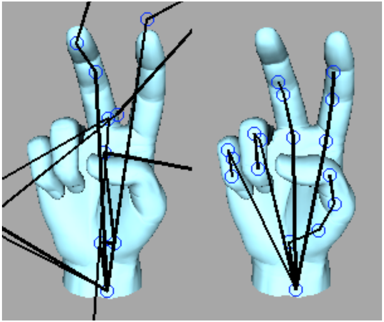


MSE for a well-textured piece of cardboard (left) and a poorly-textured surface (right).

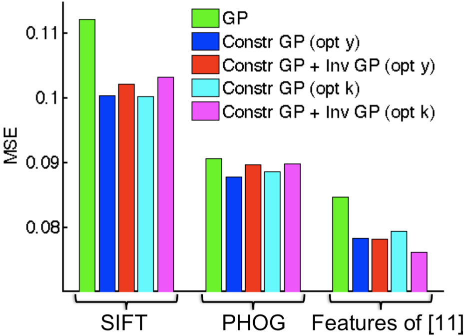
Reconstructing a Piece of Paper

Recovering the Pose of a Hand





MSE for several features.



MSE for several features.

[11] Rogez et al. CVPR'08.

Summary of constrained regression

- We proposed an effective approach to introducing constraints in discriminative methods.
- We presented a principled combination of discriminative and generative methods.
- Our framework is valid for articulated pose estimation and deformable shape reconstruction.
- We demonstrated the effectiveness of our approach in the task of hand and human body pose estimation, as well as deformable surface reconstruction.

Summary of the class

We have seen character animation

- Inverse kinematics
- NN and blending, i.e., motion graphs
- Latent variable models
- Physics (very little unfortunatelly)

Summary of the class

We have see different modules we need to choose to create our tracker

- Generative models
  - **Inference techniques:** particle filter vs optimization
  - **Likelihood models:** for monocular and multi-view settings
  - **Priors:** pose, motion, shape, physics, joint limits
- Discriminative models
  - NN
  - Regression
  - Mixture of experts
  - Subspace models
  - Structure prediction
- Combination of generative and discriminative models

More?

- Multi-view case in controlled environments is mostly solved
- Multi-view outdoors is unsolved
- Monocular tracking it's very far from been solved
- There is room for a lot of research and PhD topics.
- I'm still looking for PhD students... ;)