C280, Computer Vision

Prof. Trevor Darrell
trevor@eecs.berkeley.edu

Lecture 17: Recognition of Articulated Bodies

Last Lecture: Context

- A. Torralba, K. P. Murphy, and W. T. Freeman, "Contextual models for object detection using boosted random fields," in Advances in Neural Information Processing Systems 17 (NIPS), 2005.
- D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," in Computer Vision and Pattern Recognition, 2006
- G. Heitz and D. Koller, "Learning spatial context: Using stuff to find things," in ECCV 2008, pp. 30-43.

Today: Articulated Recognition

- Introduction / Challenges / Basic Approaches
- Regression approach
  - Urtasun and Darrell's local GP-model
- Pictorial Structures
  - Felzenswab and Huttenlocher
  - Ramanan et al.
- Strong local features
  - Lubomir Bourdev's poselets
- Strong global models
  - Black et al.'s work with the SCAPE model

People Tracking

Deva Ramanan
Toyota Technological Institute at Chicago

Is it that bad?

Probably not

Observation: 95% of the time, people + backgrounds are boring
Can track using motion priors and/or background models
Is data association solved?

What about other 5%?

Chicago White Sox
World Series

Andy Serkis's performance
Lord of the Rings

Berkeley campus

What about other 5%?

Chicago White Sox
World Series

Andy Serkis's performance
Lord of the Rings

Berkeley campus

(Perhaps) its more interesting to track

Why is finding the "people-pixels" hard?

variation in appearance

variation in pose & aspect

occlusion & clutter

Strategy 1: Pixel-based approaches
bg subtraction

- Subtract im from bg estimate
- bg estimate = known image or statistical average of history

Haritaoglu et al. PAMI00, Stauffer & Grimson, PAMI00

Strategy 1: Pixel-based approaches
fg enhancement

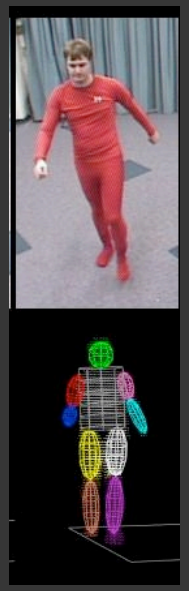
- skin detection

-Compute P(rgb|skin)
vs P(rgb|~skin)

-Tuned for Caucasians

Jones and Rehg, IJCV02
Fleck et al ECCV96

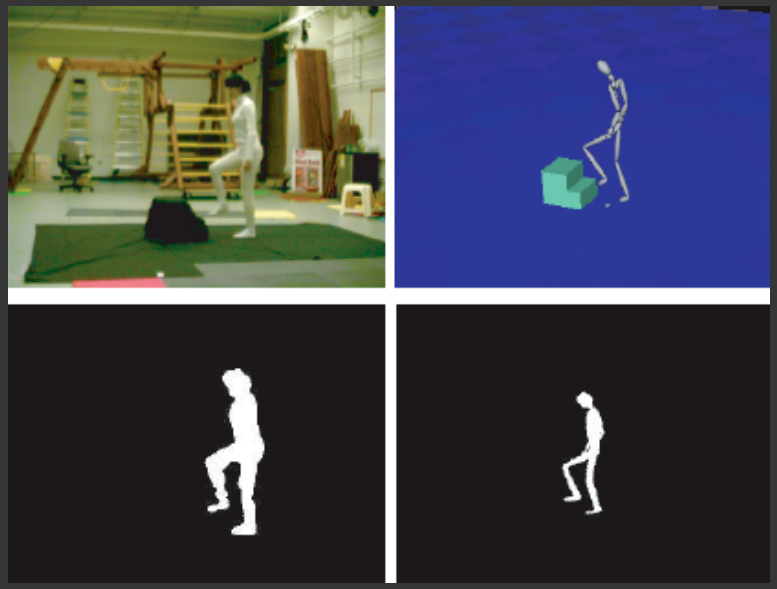
- color detection



Mikic et al CVPR01

Strategy 1: Pixel-based approaches
fg enhancement

If it can be used, it generally should be!



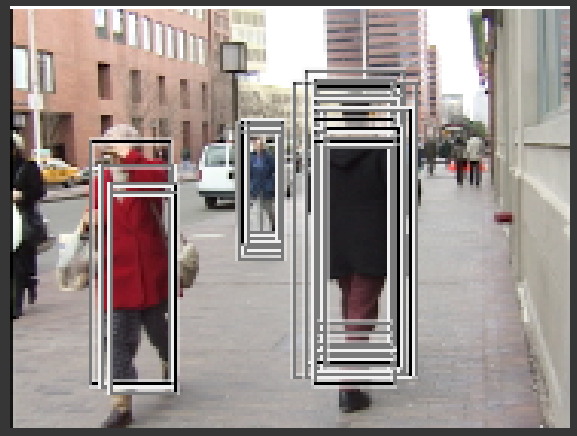
Lee et al, SIGGRAPH02

Easy to implement & reliable in controlled situations (ie, markerless motion capture)

Strategy 2: Scanning window

(+)

(-)



Papageorgiou and Poggio, ICIP99

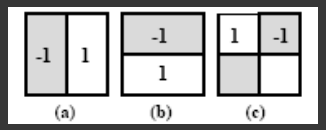
Dalal and Triggs, CVPR05

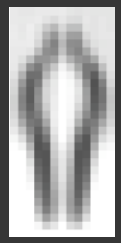
Learn pedestrian vs background
classifier from training data

Strategy 2: Scanning window features

Need invariance to appearance; focus on contours

- Haar wavelet features



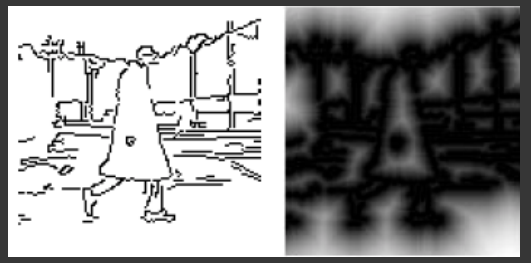


Papageorgiou and Poggio, ICIP99

- Histogram of Gradients (HOG)/SIFT descriptors

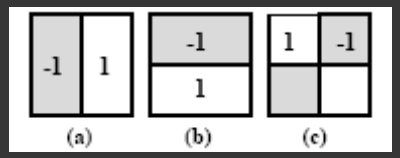
Dalal &Triggs CVPR05

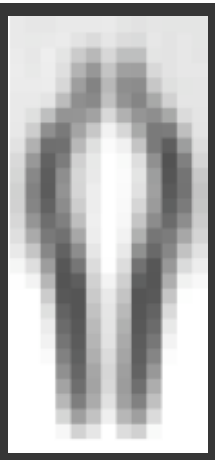
- Edges
(evaluated with chamfer score)



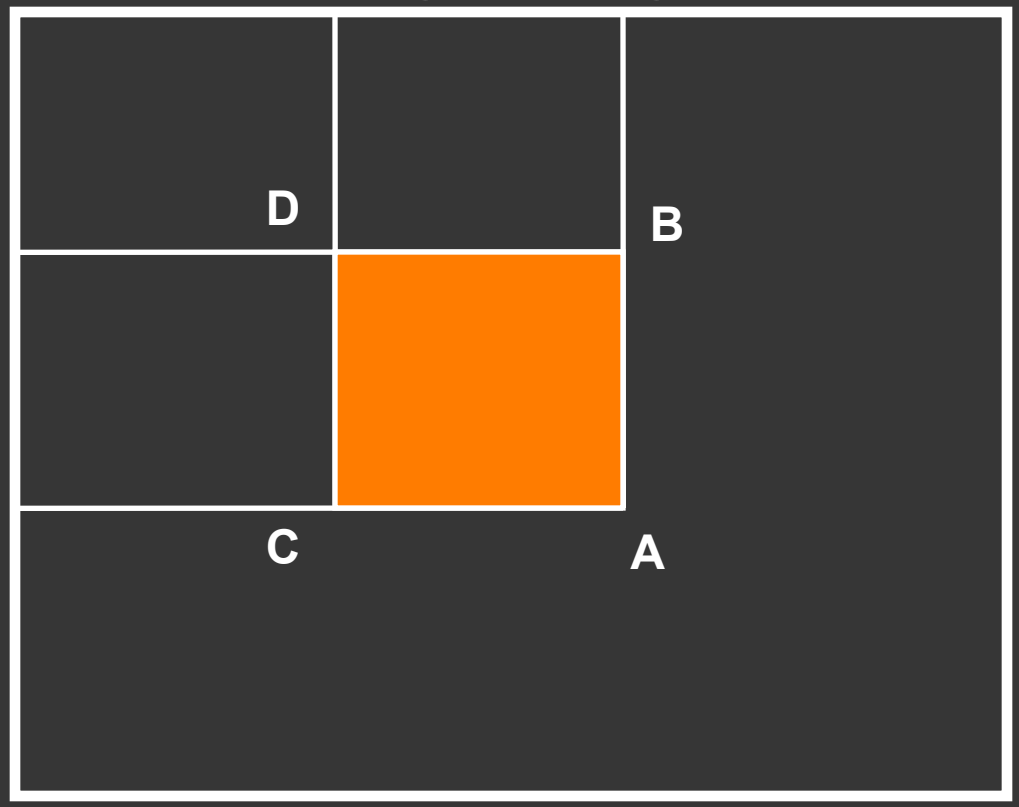
Gavrila and Philomin ICCV99

Features: Haar wavelets





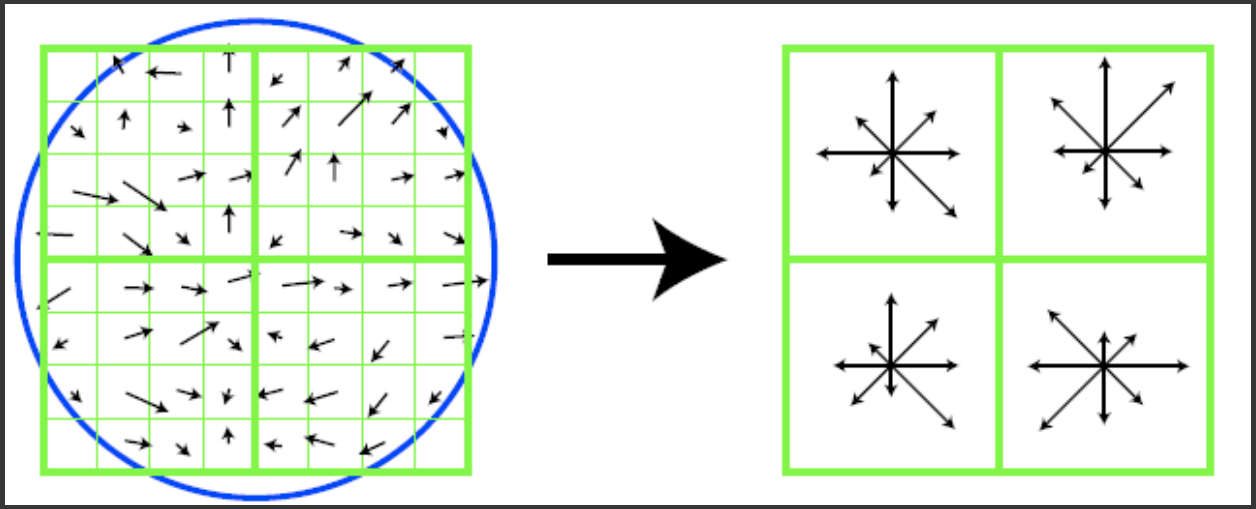
Integral Image



Sum = A - B - C +D

Viola Jones CVPR01

Features: histograms of gradients



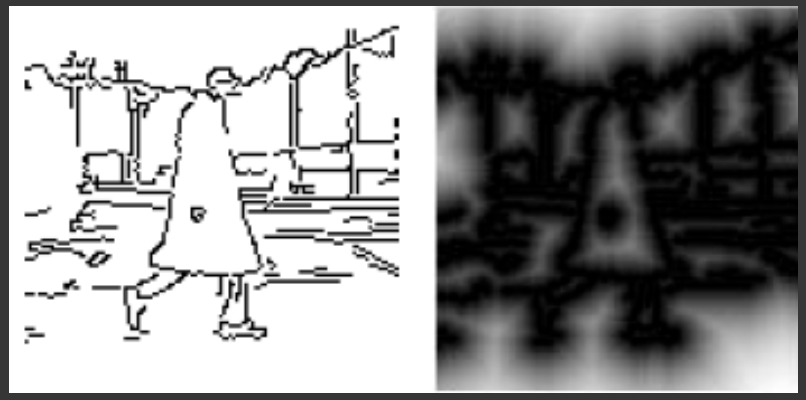
Gradients within 8X8 patch

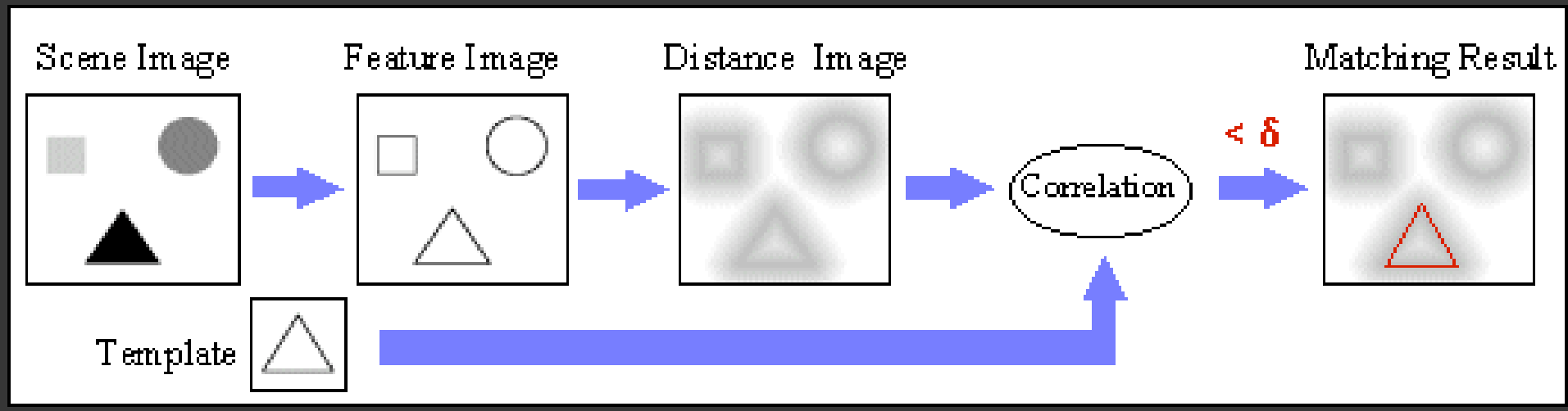
Bin into local (4X4) neighborhoods
& 8 orientations

Lowe IJCV2004
Dalal &Triggs CVPR05
Freeman and Roth IAFGR 1995

Binning achieves invariance to small patch offsets

Features: oriented chamfer edges





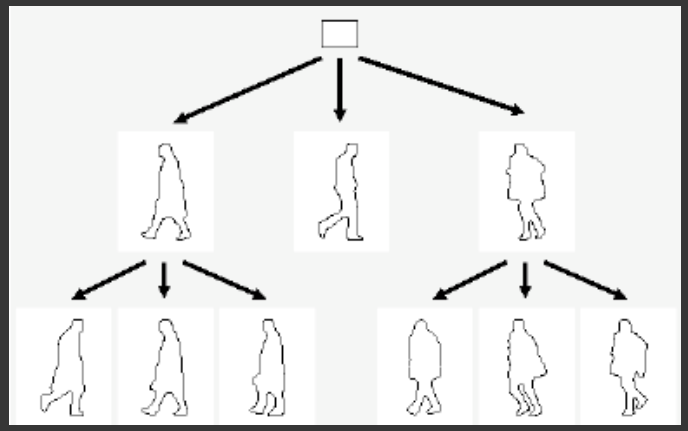
Gavrila and Philomin ICCV99

Matching can handle small deformations in the template/scene

Strategy 2: Scanning window
efficient scanning

- Coarse-to-fine search

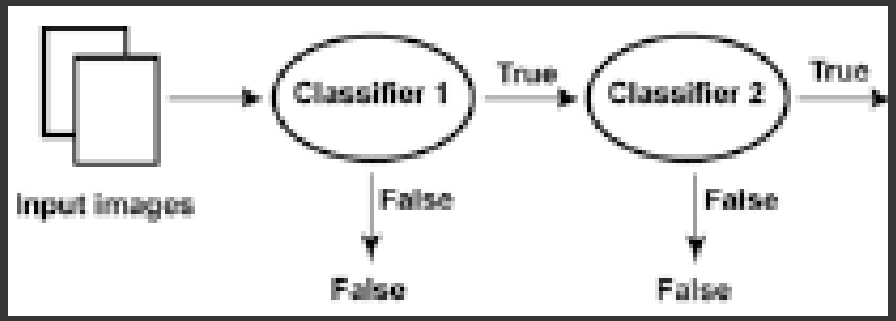
  -coarse-to-fine in both template and image domain



Gavrila and Philomin, ICCV99
Stenger et al, ICCV03

- Cascade

  -prune away most windows with initial classifier



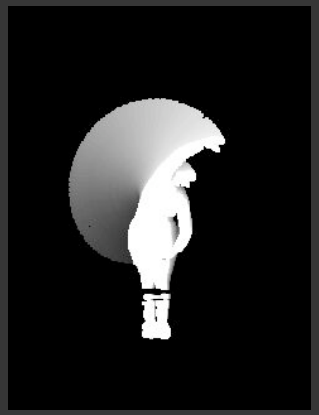
Viola and Jones CVPR01
Viola et al, ICCV03

Strategy 3: XYT window

Single frame might not be enough to find person

Motion History Image

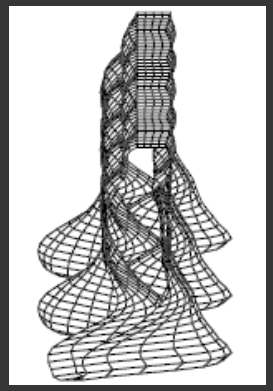
1)Do bg subtraction
2)MHI = pixel is brighter the more recently it was fg

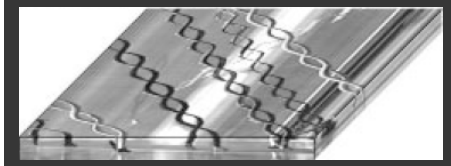


Bobick and Davis, PAMI01

Strategy 3: XYT window (cont'd)

- Look for symmetry in XYT slices

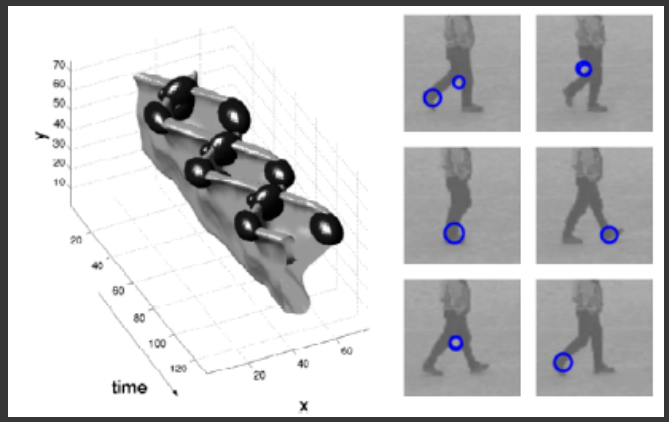




- Look for XYT interest points

Niyogi and Adelson CVPR94
Polana and Nelson, ICPR94

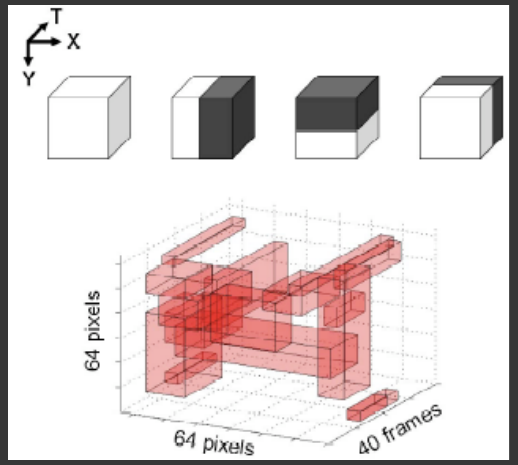
-define harris detector for XYT



Laptev and Lindeberg ICCV03

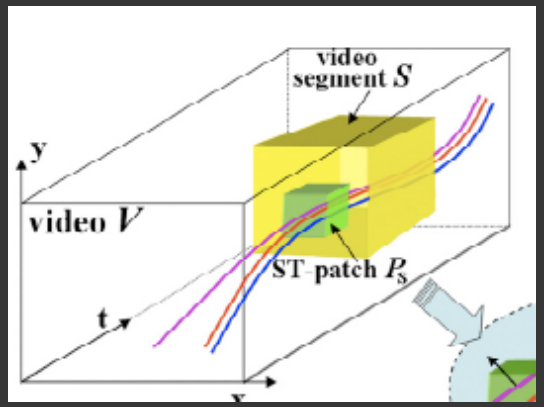
Strategy 3: XYT window (cont'd)

- Define XYT feature for classifier
  - -applied to flow
  - (invariant to appearance)



Viola et al ICCV03, Ke et al ICCV05

- Define XYT template & correlate
  - -use local flow
  - as feature



Shechtman and Irani CVPR05

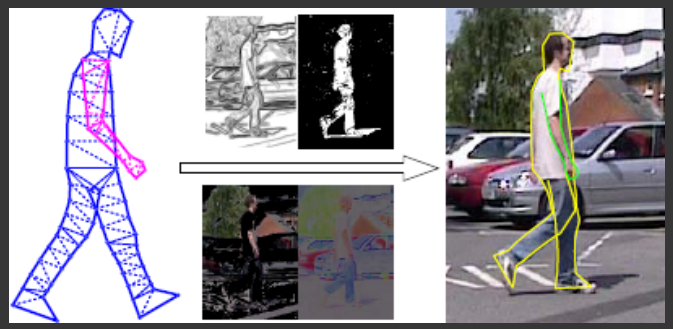
Strategy 3: XYT window (cont'd)

Shechtman and Irani CVPR05

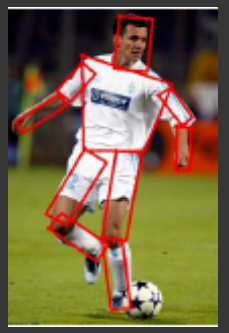
Strategy 4: Top-down pose estimation

- Compute $P(\Theta|I) \alpha P(I|\Theta)P(\Theta)$ by sampling methods
  - Iteratively search space of body poses Θ
    - sample from prior or data-driven proposal
  - Works well with informative likelihood (skin) and/or prior (walking)

Lee & Cohen CVPR04



Zhang et al, CVPR04

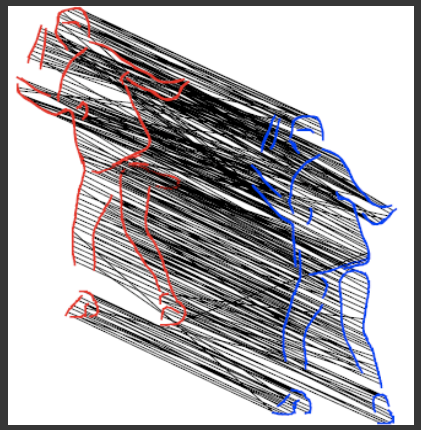


Hua et al CVPR05

Strategy 4: Top-down pose estimation (cont'd)

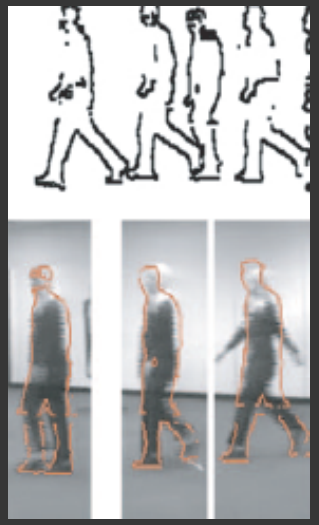
- Match exemplars
  - Encode articulations by templates or on-the-fly deformations
  - Seems to be limited to standard poses (useful for tracker initialization)

Model template



Query edge map

Sullivan & Carlsson, ECCV02
Loy et al ECCV04
Mori & Malik ECCV02



Gavrila & Philomin, ICCV99
Toyama & Blake ICCV01

- Efficient search
  - Coarse to fine
  - Approx. Nearest Neighbors (Shakhnarovich et al ICCV03)

Local Probabilistic Regression for Activity-Independent Human Pose Inference

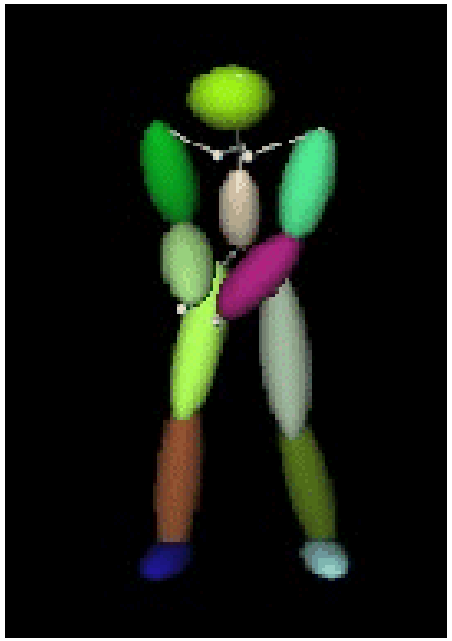
Raquel Urtasun and Trevor Darrell

Human pose inference

Regression approach to human pose inference: map visual observations to articulated body configurations.

Goals:

- Deal with high dimensional input and output spaces
- Deal with arbitrary motions that require very large training sets, e.g., 10^6
- Efficiency: work in real time
- Learn multi-modal mappings, i.e., an input has more than one output
- Provide reliable confidence of the learned mapping
- Prune redundant examples
- Provide guaranties in the approximation



Regression approaches to pose estimation and tracking

- Neural Networks [*Rosales & Sclaroff 00*]
- RVM [*Agarwal and Triggs 04, 07*]
- Bayesian mixture of experts [*Sminchisescu et al. 05, 08*]

training set size N <= 3000

→ Very limited set of activities;

- Nearest Neighbor techniques [*Efros et al. 03*, *Shakhnarovich et al. 03*].

Can deal with large training sets but

→ No algorithm for pruning

→ No confidence measure

→ Can have poor generalization

Our approach extends GPs to deal with arbitrary large datasets and model multimodal mappings

Gaussian Processes Regression

Given a training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i), i = 1, \cdots, N\}$ composed of inputs $\mathbf{x}_i$ and outputs, $\mathbf{y}_i$ we have

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

Bayesian approach that assumes a GP prior over functions such that

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{i=1}^{D} N(\mathbf{Y^{(i)}}; \mathbf{0}, \mathbf{K})$$

where $\mathbf{Y} = (\mathbf{y}_1, \cdots, \mathbf{y}_N)^T \quad \mathbf{Y} \in \Re^{N \times D}$

$\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_N)^T \quad \mathbf{X} \in \Re^{N \times Q}$

Non-parametric, entirely defined by the covariance, $\mathbf{K} \in \Re^{N \times N}$,e.g., RBF

Very large computational cost $\mathcal{O}(N^3)$ and memory requirements $\mathcal{O}(N^2)$

Sparsification

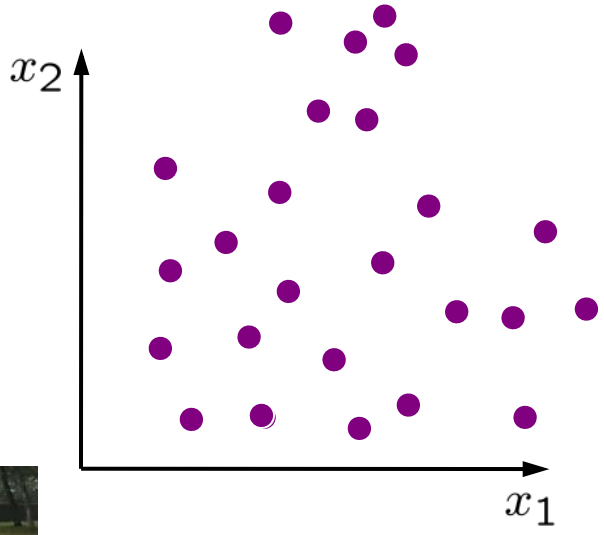
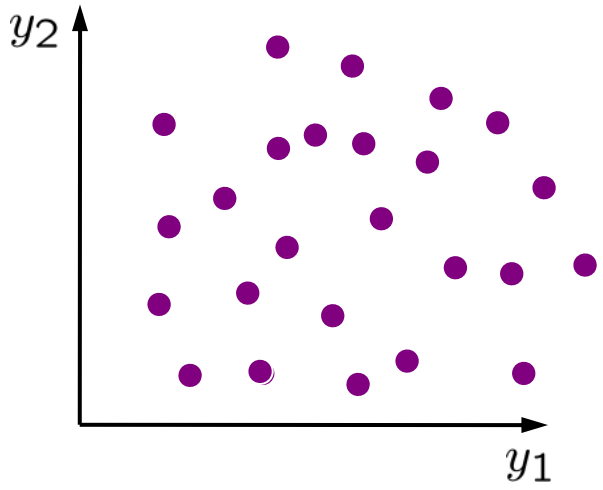
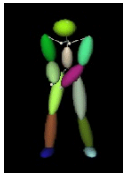


Image feature space ($\mathbf{x}$)



Pose space ($\mathbf{y}$)



Subset of the Data (SOD)

Sparsification

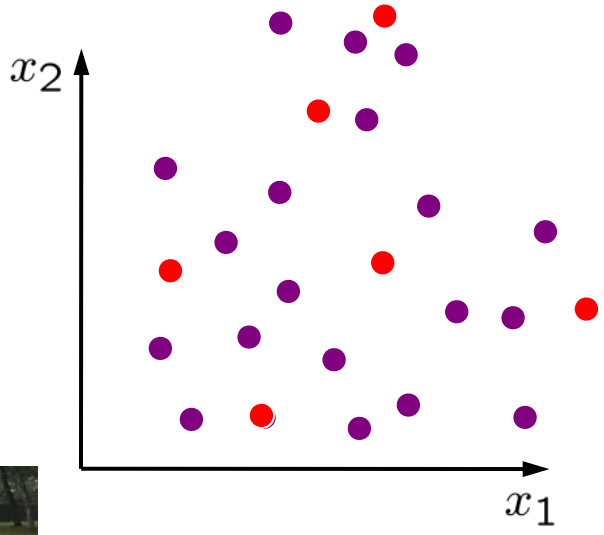
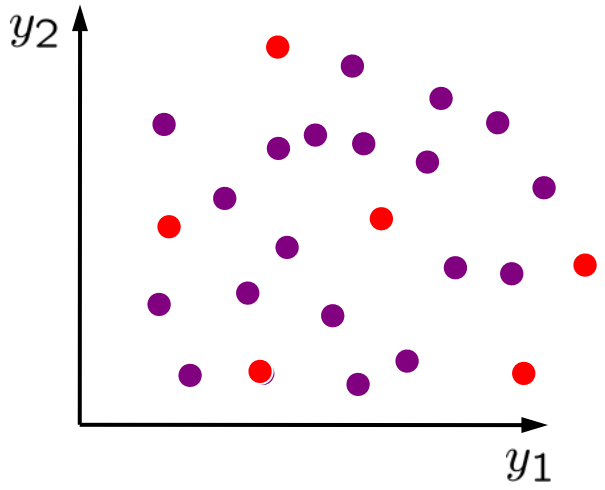
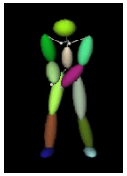


Image feature space ($\mathbf{x}$)



Pose space ($\mathbf{y}$)



Subset of the Data (SOD)

Sparsification

Smaller computational cost $\mathcal{O}(m^3)$ and memory requirements $\mathcal{O}(m^2)$

If very large database, $m \ll N$ and bad approximation

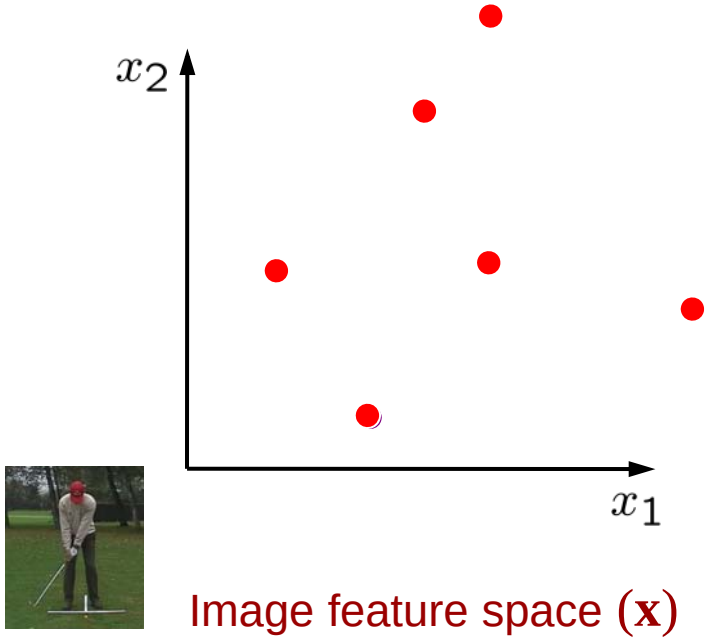
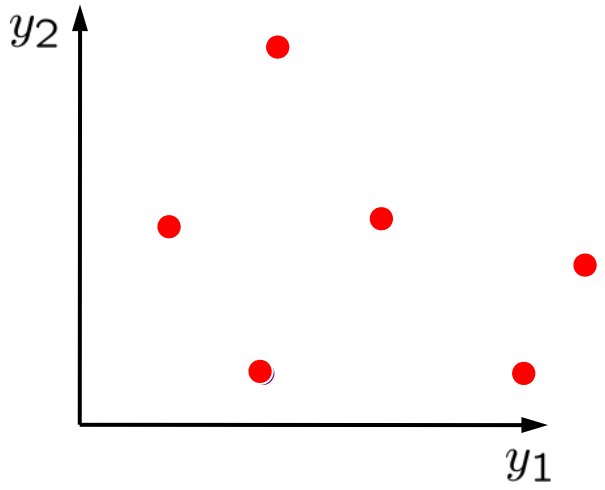
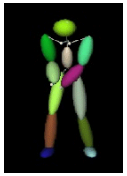


Image feature space ($\mathbf{x}$)





Pose space ($\mathbf{y}$)

Subset of the Data (SOD)

Sparsification

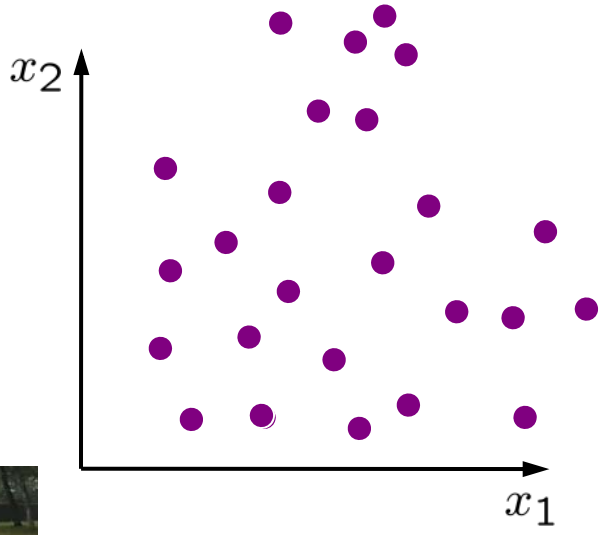
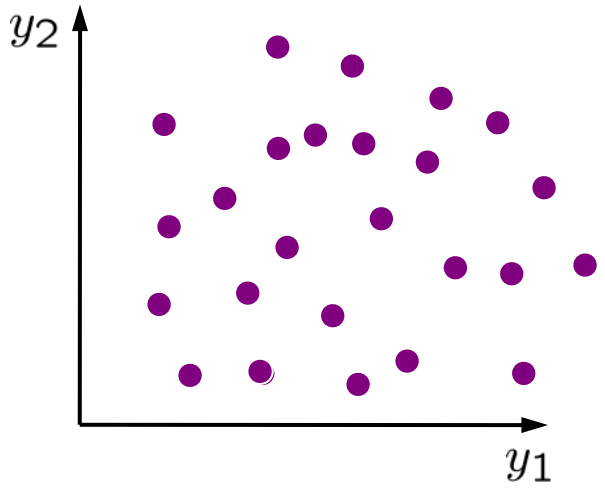
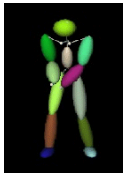


Image feature space ($\mathbf{x}$)





Pose space ($\mathbf{y}$)

Apply inducing variables…

Sparsification

Incorporate additional variables for approximation

Assume independency between training and testing, so that inference only depends on the inducing variables, not on the training data

If very large database, $m \ll N$ and bad approximation

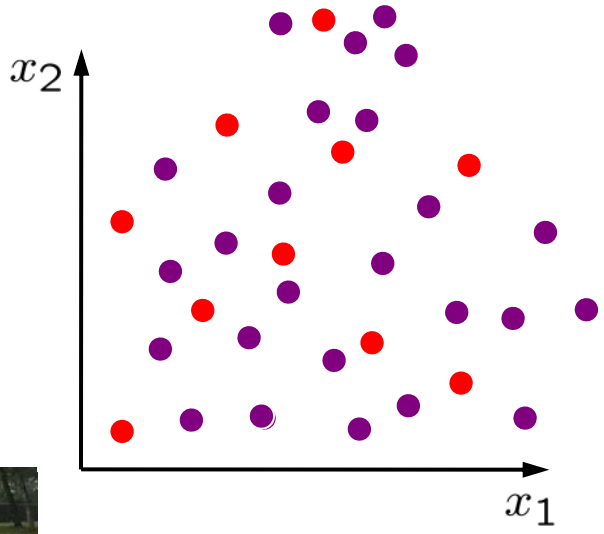
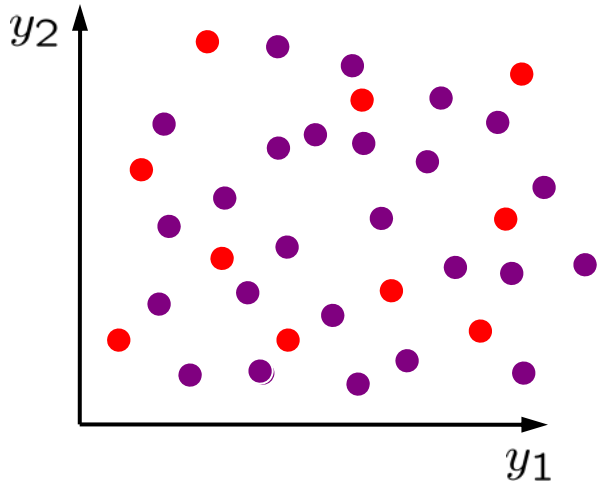
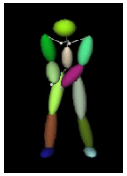


Image feature space ($\mathbf{x}$)



Pose space ($\mathbf{y}$)



Apply inducing variables…

Sparsification

Different philosophy, don't eliminate training data

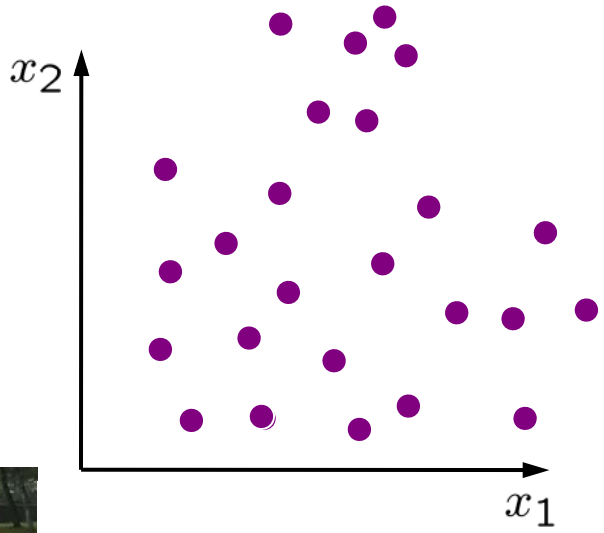
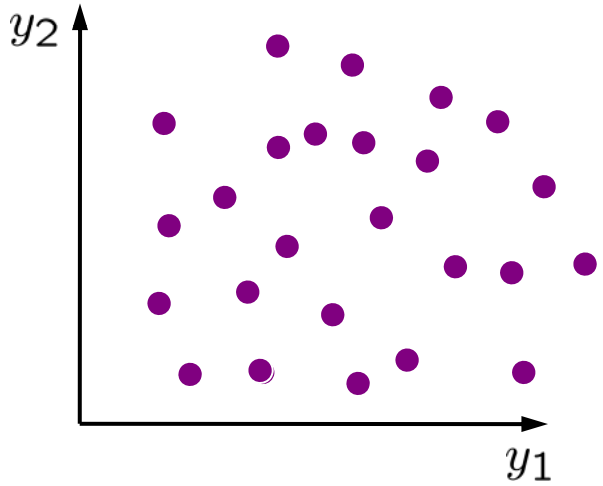
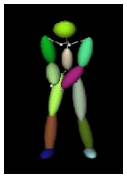


Image feature space ($\mathbf{x}$)





Pose space ($\mathbf{y}$)

Local approximations: Static sparsification

Sparsification

Different philosophy, don't eliminate training data

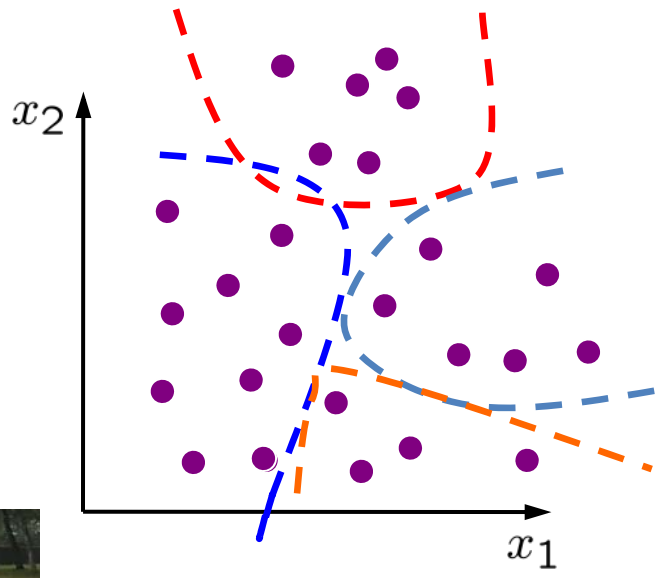
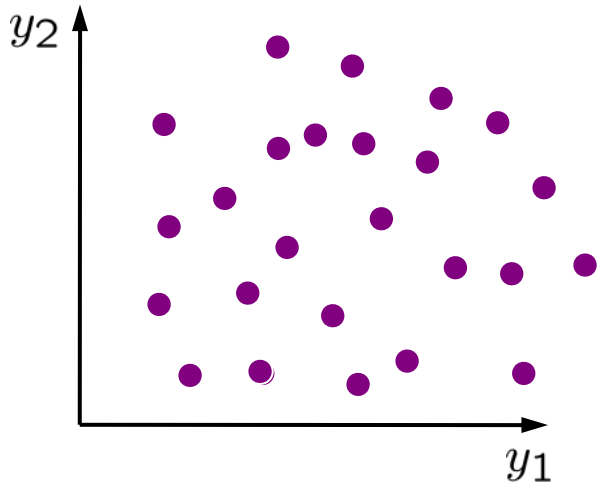
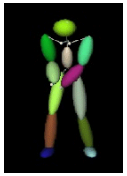


Image feature space ($\mathbf{x}$)



Pose space ($\mathbf{y}$)



Local approximations: Static sparsification

Sparsification

Different philosophy, don't eliminate training data

Bad approximation in the boundaries between clusters and where the mapping is multimodal…

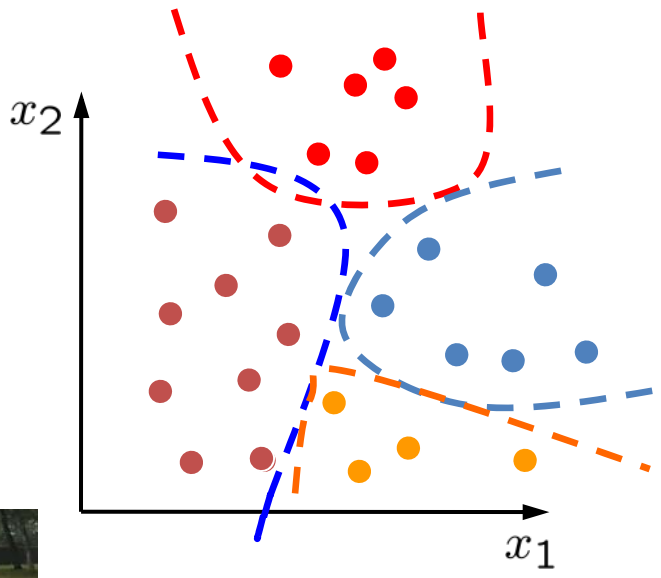
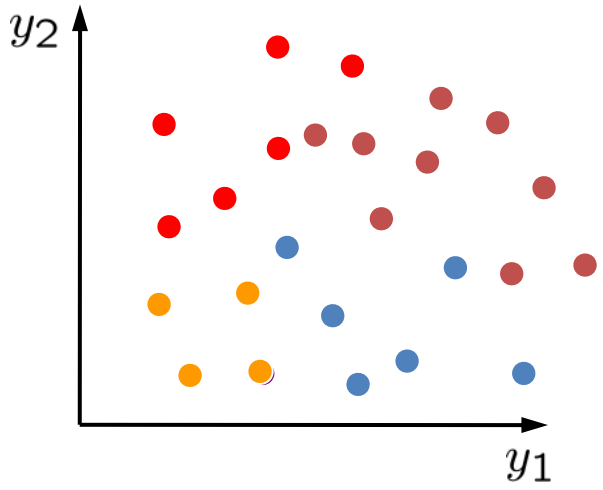
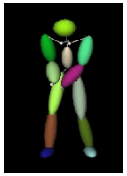


Image feature space $(\mathbf{x})$



Pose space $(\mathbf{y})$



Local approximations: Static sparsification

Online Sparsification

Local approximation to the regression, assumes monotonically decreasing covariance functions

No boundary problem for inference since regression is centered at the test

ADVANTAGE: Computational time is much smaller than before $\mathcal{O}(Sm^3)$ this is the same as before, but now m can be much smaller!

Inference is very simple:

$$\mu(\mathbf{x}_*) \approx K_{*,\zeta}(\mathbf{K}_{\zeta,\zeta} + \sigma^2_{noise}\mathbf{I})^{-1}\mathbf{Y}_\zeta$$
$$\sigma(\mathbf{x}_*) \approx k_{*,*} - K_{*,\zeta}(\mathbf{K}_{\zeta,\zeta} + \sigma^2_{noise}\mathbf{I})^{-1}K_{\zeta,*}$$

where ζ is the neighborhood of each test point. Each test point is independent

Searching the neighbors has $\mathcal{O}(N)$ → Use LSH, Trees, etc.

Algorithm

Offline phase: Learn hyper-parameters offline for a subset of local GPs

Online phase: Inference with a local mixture

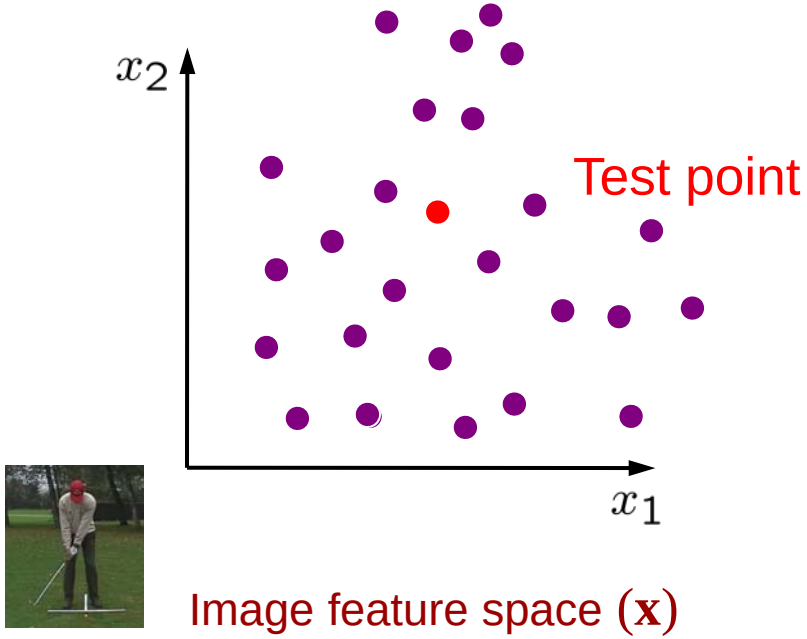
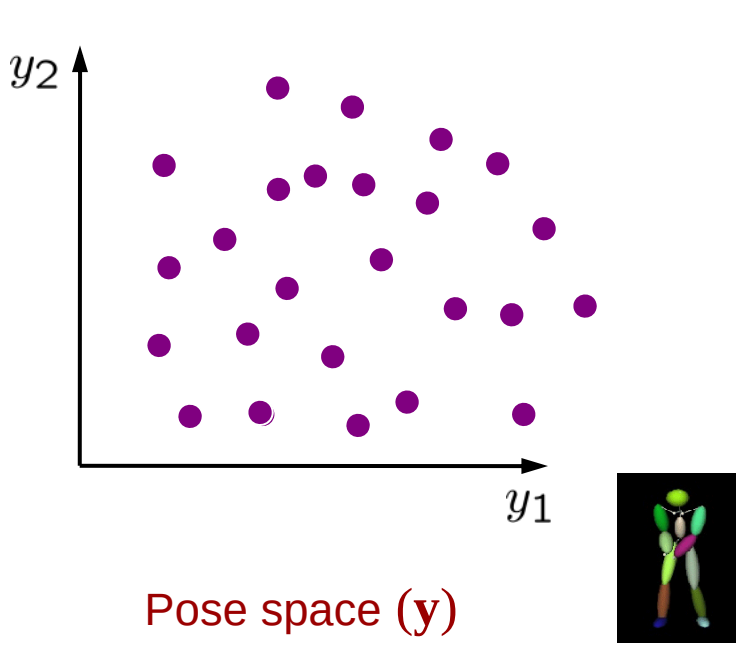


Image feature space $(\mathbf{x})$



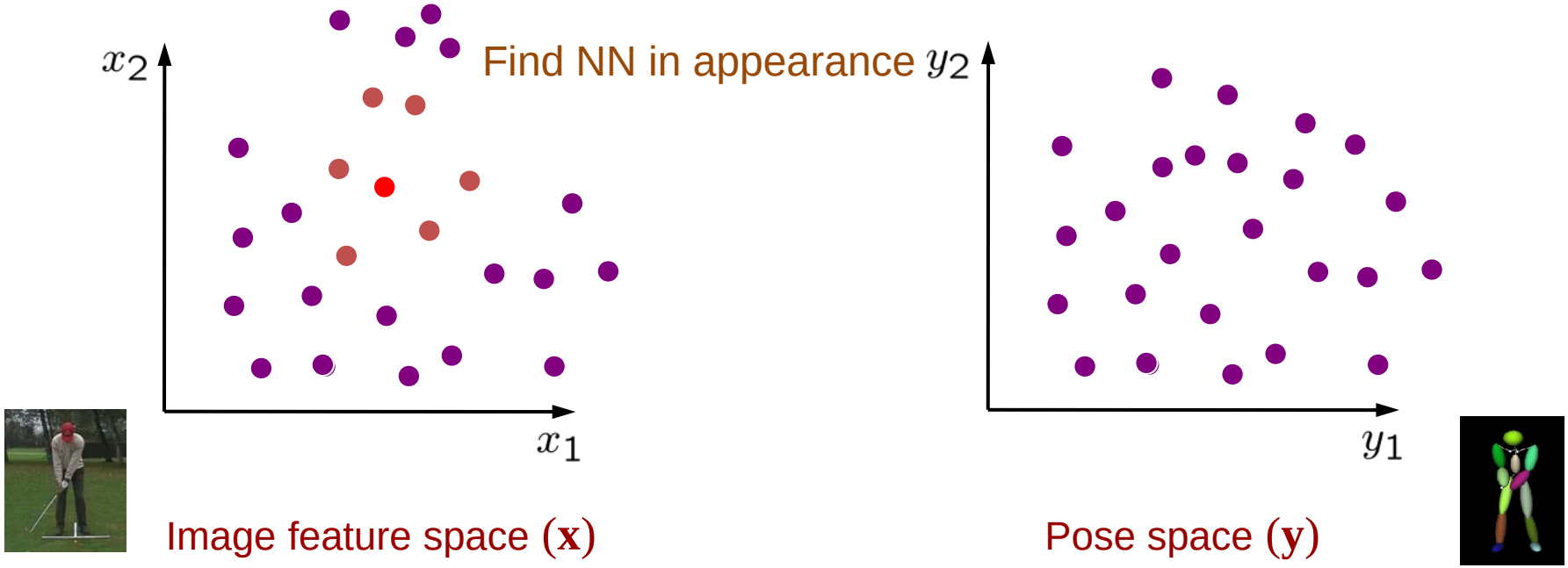
Pose space $(\mathbf{y})$

[Urtasun and Darrell, CVPR 2008]

Algorithm

Offline phase: Learn hyper-parameters offline for a subset of local GPs

Online phase: Inference with a local mixture



[Urtasun and Darrell, CVPR 2008]

Algorithm

Offline phase: Learn hyper-parameters offline for a subset of local GPs

Online phase: Inference with a local mixture

They are not necessary NN in pose

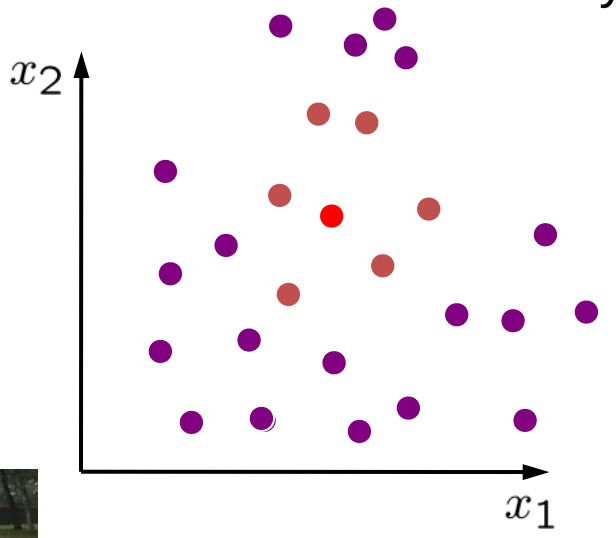
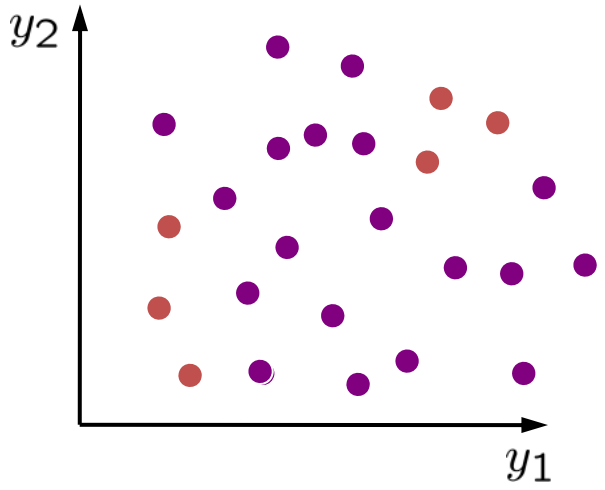
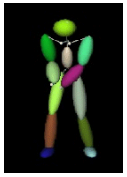


Image feature space $(\mathbf{x})$





Pose space $(\mathbf{y})$

[Urtasun and Darrell, CVPR 2008]

Algorithm

Offline phase: Learn hyperparameters offline for a subset of local GPs

Online phase: Inference with a local mixture

For each NN in appearance, define a local GP in pose space → avoid multimodality

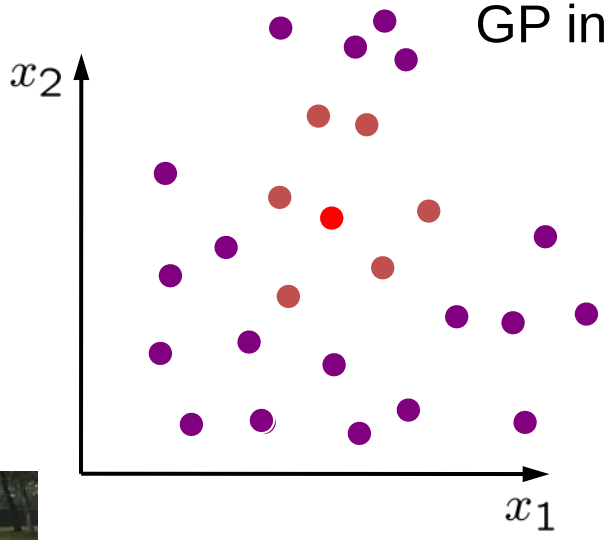
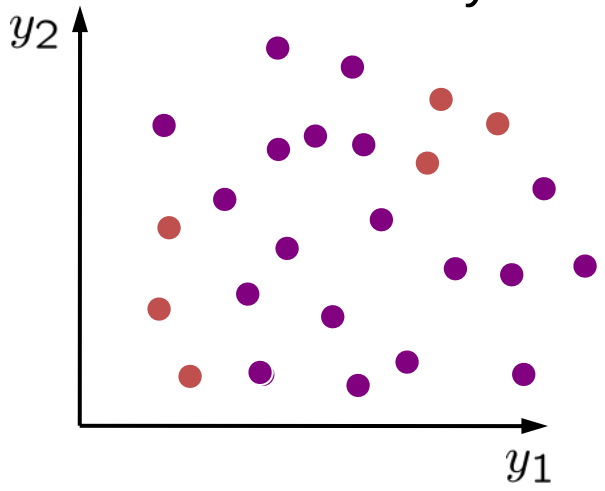
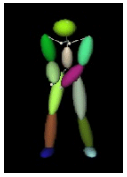


Image feature space $(\mathbf{x})$



Pose space $(\mathbf{y})$



[Urtasun and Darrell, CVPR 2008]

Algorithm

Offline phase: Learn hyperparameters offline for a subset of local GPs

Online phase: Inference with a local mixture

For each NN in appearance, define a local GP in pose space → avoid multimodality

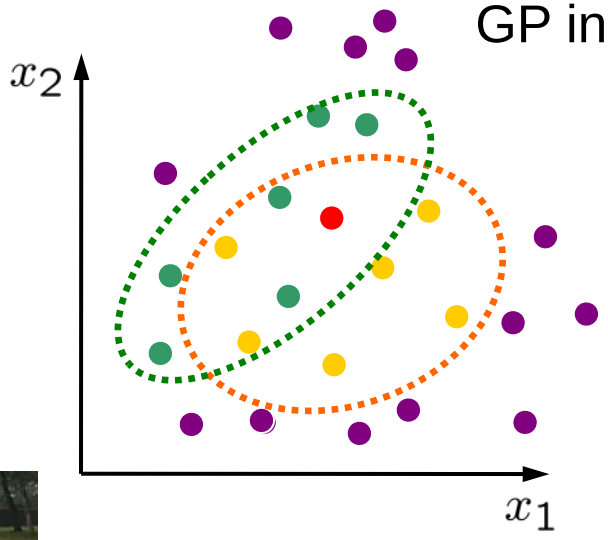
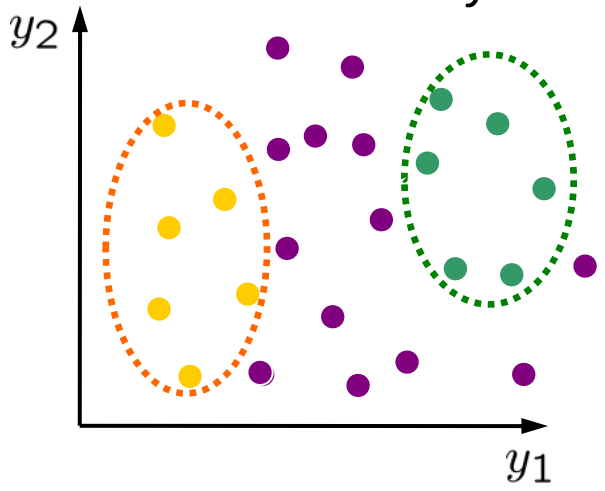
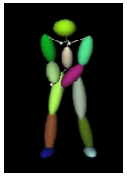


Image feature space $(\mathbf{x})$





Pose space $(\mathbf{y})$

Essentially a probabilistic GP version of locally weighted regression, with bound on the increment of uncertainty for monotonically decreasing covariance functions and unimodal mappings

[Urtasun and Darrell, CVPR 2008]

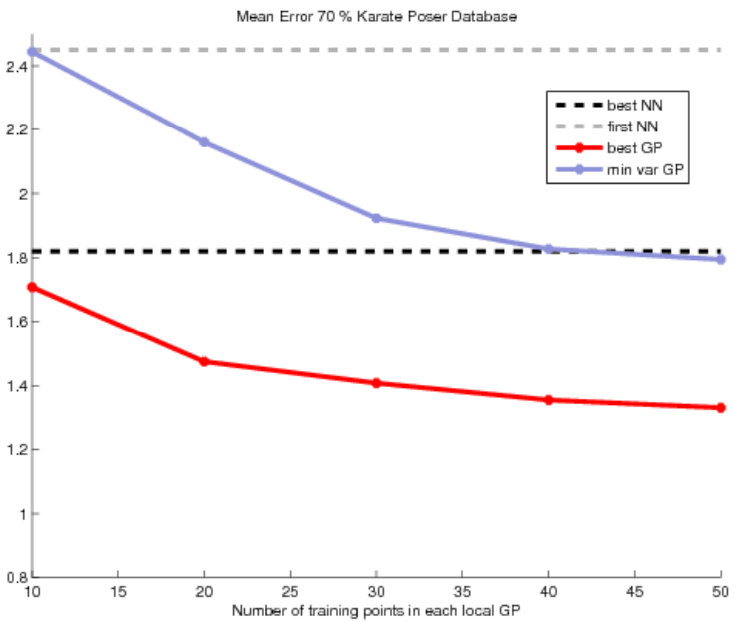
Experiments: From small to large databases

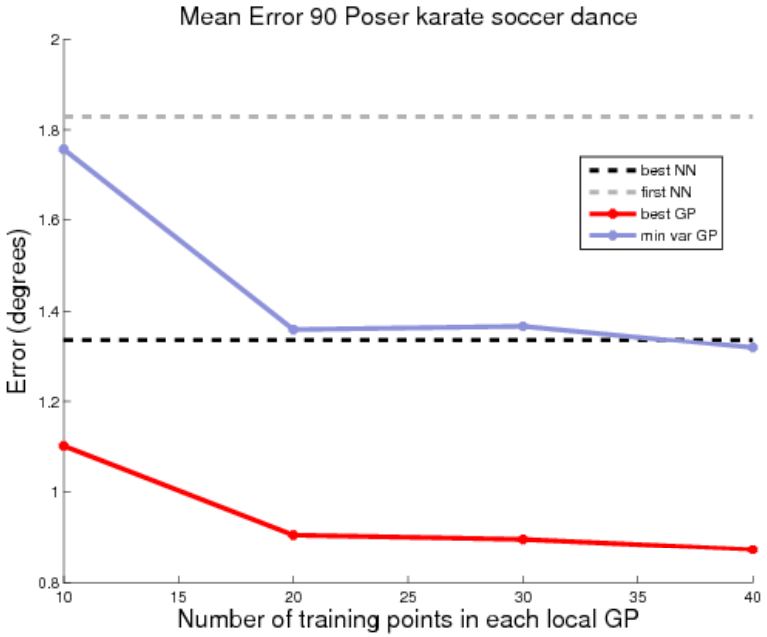
DB size	1-NN	Best of-10-NN	GP $(S = 10)$	GP $(S = 20)$	GP $(S = 30)$	GP $(S = 40)$
1,500	0.88 ± 1.77	0.71 ± 1.38	0.83 ± 1.53	0.98 ± 1.70	$\mathbf{0.56 \pm 1.40}$	0.70 ± 1.45
15,000	1.92 ± 2.76	1.49 ± 1.81	1.32 ± 2.07	1.10 ± 1.88	1.03 ± 1.81	$\mathbf{0.99 \pm 1.77}$
50,000	1.83 ± 2.62	1.34 ± 1.47	1.10 ± 1.85	0.91 ± 1.64	0.90 ± 1.66	$\mathbf{0.87 \pm 1.58}$

*Our algorithm **works in all scenarios**; with a large or small number of examples. A small neighborhood (S<=40) is sufficient to produce very accurate results. 3D mean error (in degrees) in a database generated using Poser from Mocap data.*

[Urtasun and Darrell, CVPR 2008]

Experiments: Very large databases



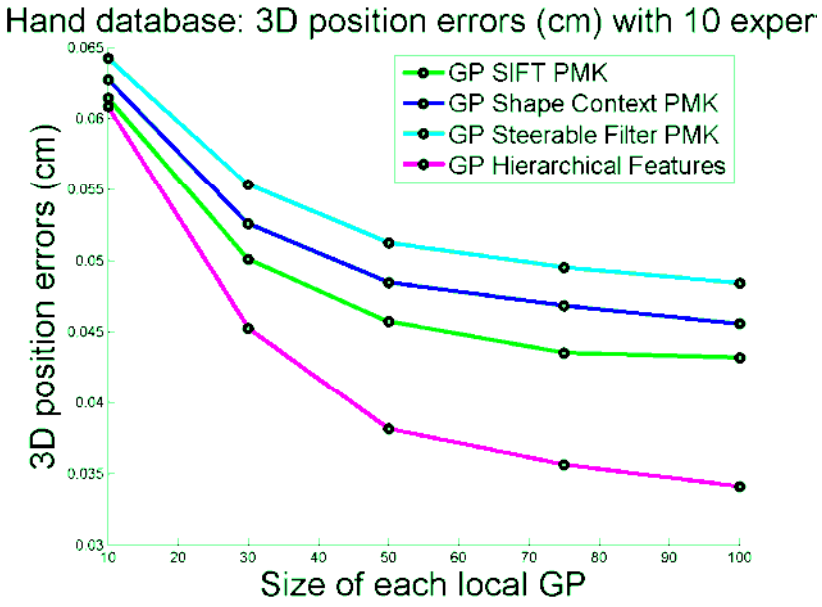


Inference in very large datasets:

(left) 15,000 and (right) 50,000 example Poser database composed of multiple activities: karate, soccer, dance, etc. The mean errors were smaller than 1 degree.

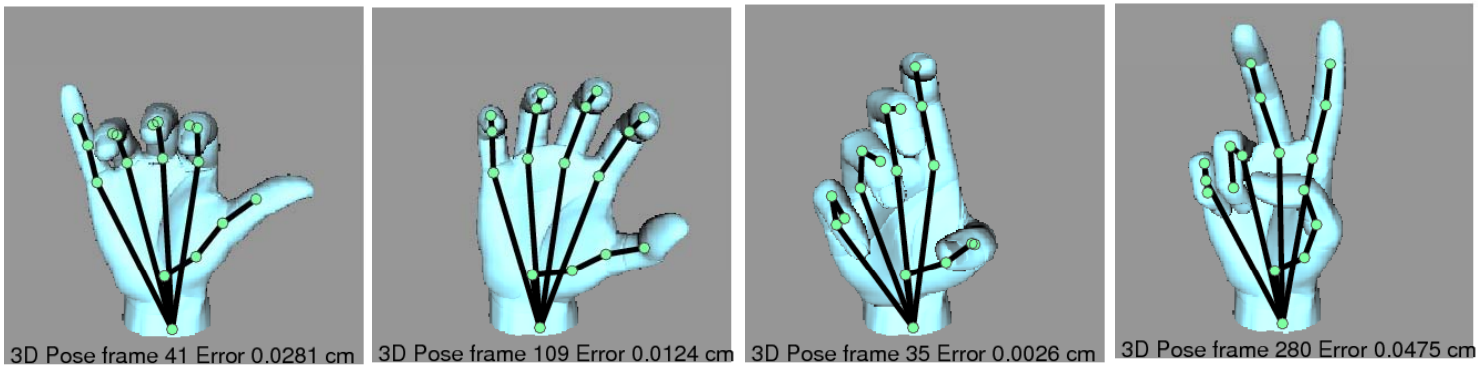
[Urtasun and Darrell, CVPR 2008]

Experiments: Different Kernels



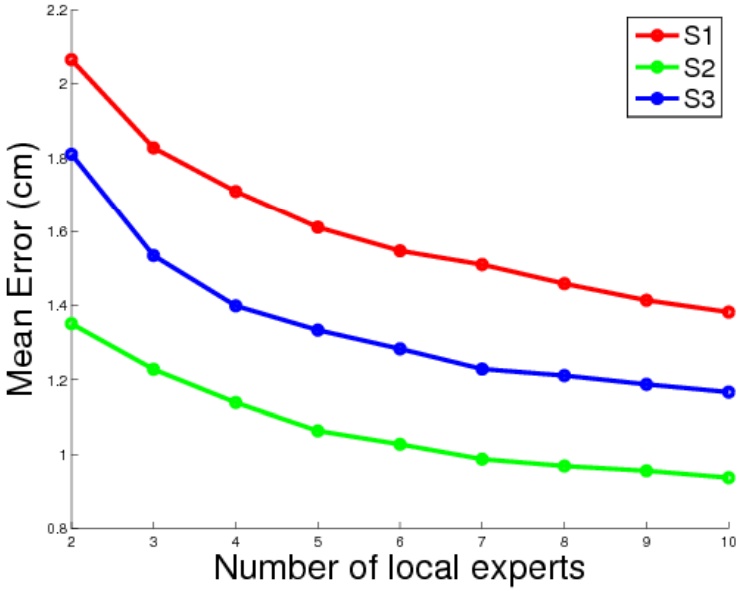
***Generalization to Different Kernels**: our mixture of local online GPs provides accurate results with kernels based on different features. Mean prediction errors for a hand database when using Pyramid Match kernels based on SIFT, Shape Context,Steerable filters, or Hierarchical features are shown.*

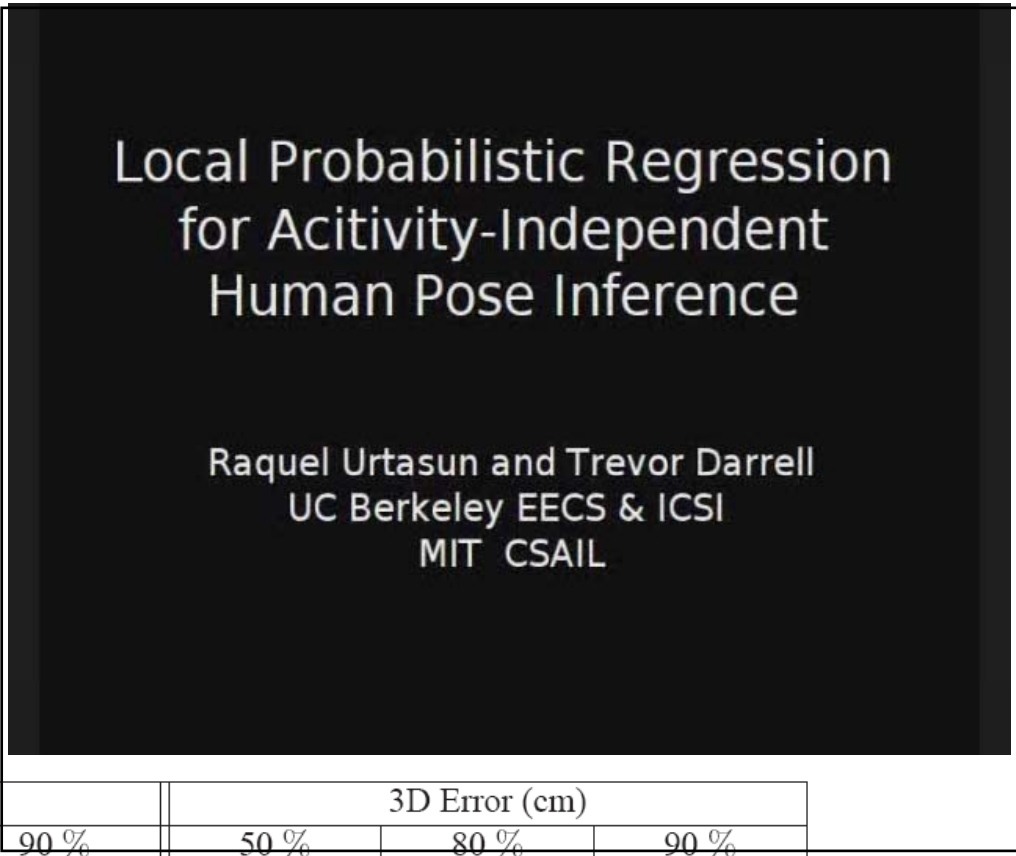
Examples of 3D reconstruction



[Urtasun and Darrell, CVPR 2008]

Experiments: Real Data





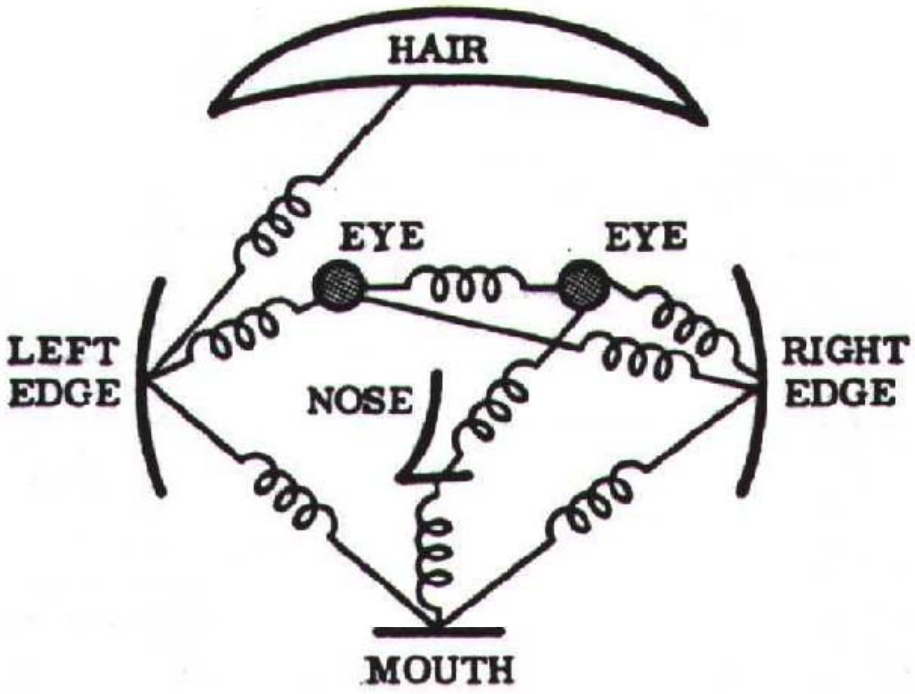
	2D Error (pixels)			3D Error (cm)		
% database	50 %	80 %	90 %	50 %	80 %	90 %
S1	2.01 ± 1.80	1.44 ± 1.46	1.28 ± 1.35	1.37 ± 1.06	0.94 ± 0.90	0.91 ± 0.97
S2	1.22 ± 1.05	0.87 ± 0.85	0.80 ± 0.86	0.93 ± 0.65	0.68 ± 0.55	0.60 ± 0.51
S3	1.67 ± 1.69	1.33 ± 1.57	1.24 ± 1.76	1.16 ± 0.91	0.94 ± 0.81	0.88 ± 0.89

Humaneva dataset: *Mean errors in (cm) and (pixels) when using different percentages of the database for training and testing. Our approach accurately estimates the pose, with maximum errors of 1 cm and 2 pixels. In this database the size of human figures ranged from 200 to 300 pixels.*

[Urtasun and Darrell, CVPR 2008]

Pictorial Structures Revisited…

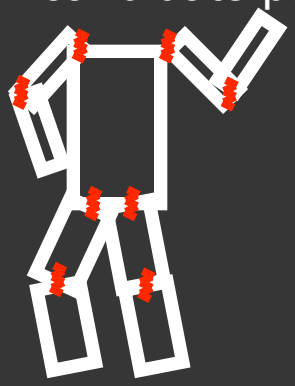
- Fischler & Elschlager 1973
- Model has two components
  - parts (2D image fragments)
  - structure (configuration of parts)

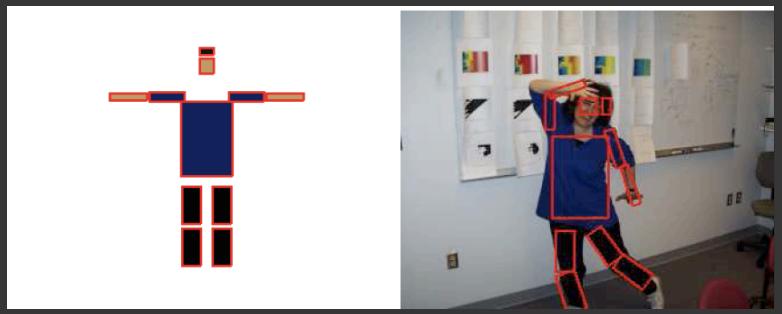


Strategy (5)
Bottom-up parts: assembly

Detect parts & then assemble

- Dynamic programming (tree model)
  - For N candidate parts $O(N^2)$, but can speed up to O(N) with distance transform

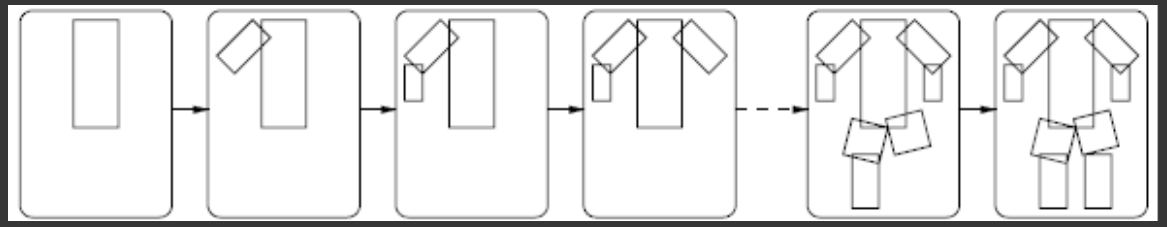




Felzenszwalb & Huttenlocher, CVPR00

Ioffe & Forsyth, ICCV01

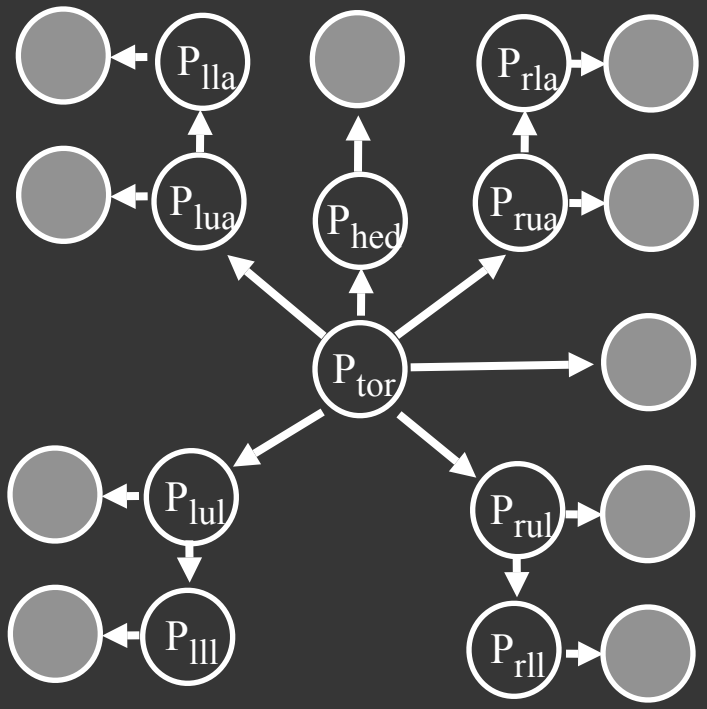
- Iteratively sample good assemblies

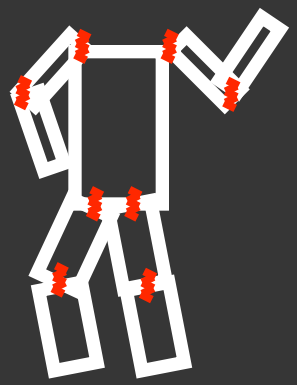


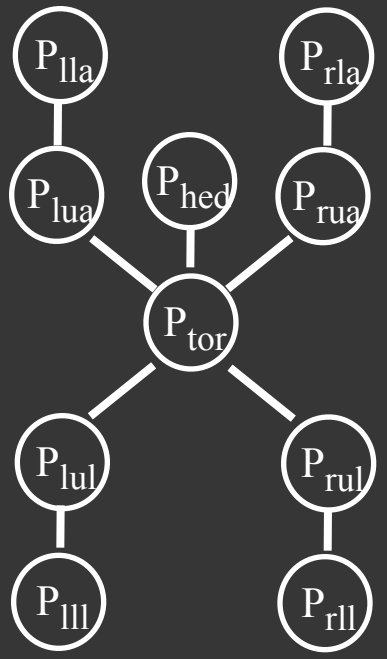
Ioffe & Forsyth, ICCV99

Pictorial structure model

Fischler and Elschlager(73), Felzenszwalb and Huttenlocher(00)







$$\Pr(P_{tor}, P_{arm}, \ldots | \mathrm{Im}) \propto \prod_{i,j} \Pr(P_i | P_j) \prod_i \Pr(\mathrm{Im}(P_i))$$

part geometry

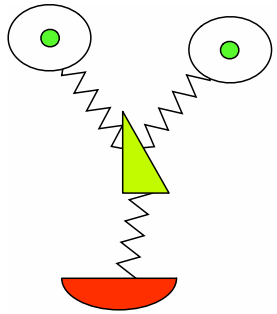
part appearance

Object Recognition with Pictorial Structures

Pedro F. Felzenszwalb
University of Chicago
pff@cs.uchicago.edu

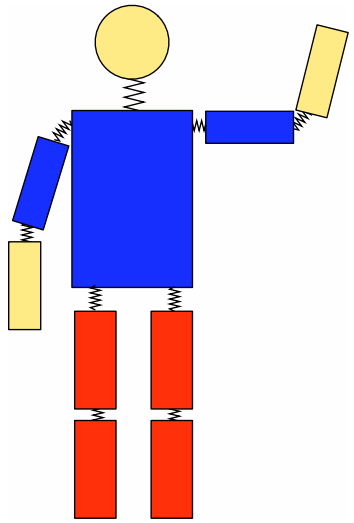
Joint work with Daniel P. Huttenlocher

Pictorial structures



Part-based representation:

- Each part models local visual properties.
- "Springs" model spatial relationships.
- Joint estimation of part locations.
  - No hard detection of parts or features.
  - No initialization parameters.



- Model is represented by a graph $G = (V, E)$.
  - $V = \{v_1, \ldots, v_n\}$ are the parts.
  - $(v_i, v_j) \in E$ indicates a connection between parts.
- $m_i(l_i)$ is the cost of placing part i at location l_i.
- $d_{ij}(l_i, l_j)$ is a deformation cost.
- Optimal location for object is given by $L^* = (l_1^*, \ldots, l_n^*)$,

$$L^* = \operatorname*{argmin}_L \left(\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i, v_j) \in E} d_{ij}(l_i, l_j) \right)$$

Efficient minimization

$$L^* = \arg\min_{L} \left(\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j)\in E} d_{ij}(l_i, l_j) \right)$$

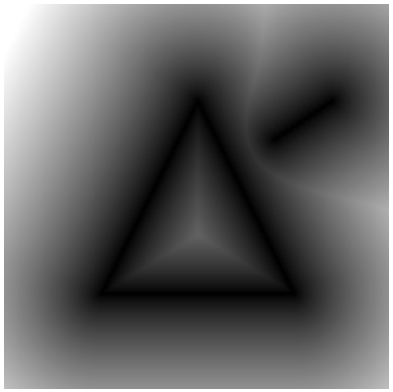
- n parts and h locations gives h^n configurations.
- If graph is a tree we can use dynamic programming.
  - $O(nh^2)$, much better but still slow.
- If $d_{ij}(l_i, l_j) = ||T_{ij}(l_i) - T_{ji}(l_j)||^2$ can use DT.
  - $O(nh)$, as good as matching each part separately!!

Distance transform

Given a set of points on a grid $P \subseteq \mathcal{G}$,
the quadratic distance transform of P is,

$$\mathcal{D}_P(q) = \min_{p \in P} ||q - p||^2$$

P



$\mathcal{D}_P$

Generalized distance transform

Given a function $f : \mathcal{G} \rightarrow \mathbb{R}$,

$$\mathcal{D}_f(q) = \min_{p \in \mathcal{G}} \left(||q - p||^2 + f(p) \right)$$

– for each location q, find nearby location p with $f(p)$ small.

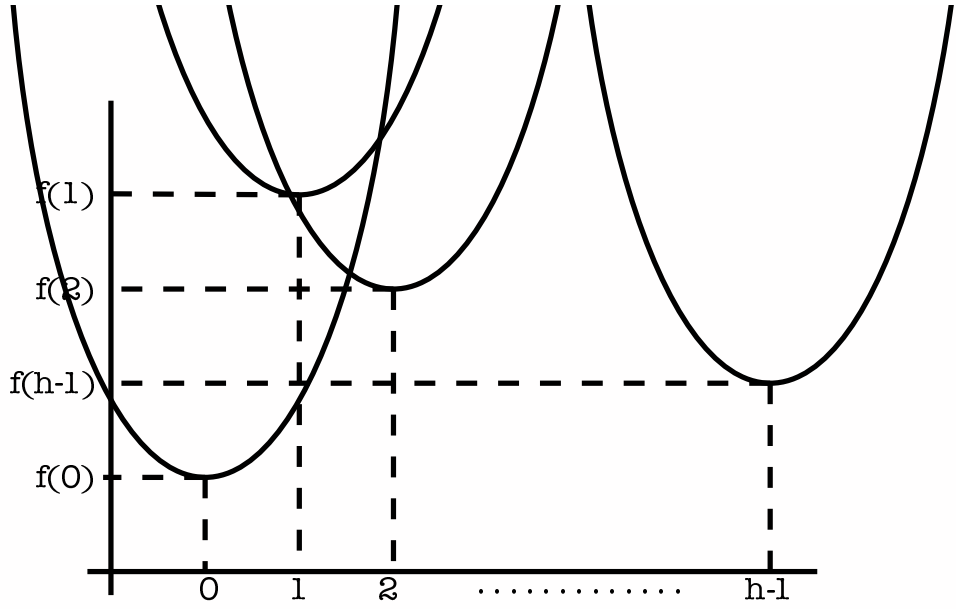
– equals DT of points P if f is an indicator function.

$$f(p) = \begin{cases} 0 & \text{if } p \in P \\ \infty & \text{otherwise} \end{cases}.$$

1D case: $\mathcal{D}_f(q) = \min_{p \in \mathcal{G}} \left((q-p)^2 + f(p) \right)$

For each p, $\mathcal{D}_f(q)$ is below the parabola rooted at $(p, f(p))$.

$\mathcal{D}_f(q)$ is defined by the lower envelope of h parabolas.



There is a simple geometric algorithm that computes $\mathcal{D}_f(p)$ in $O(h)$ time for the 1D case.

– similar to Graham's scan convex hull algorithm.

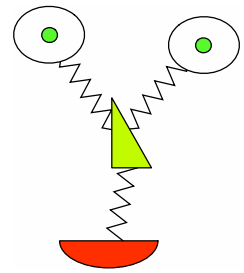
– about 20 lines of C code.

The 2D case is "separable", it can be solved by sequential 1D transformations along rows and columns of the grid.

See **Distance Transforms of Sampled Functions**, Felzenszwalb and Huttenlocher.

Simple face model

- Locations are positions in the image grid.
- Match cost $m_i(l_i)$ for placing part i at l_i.
- Central part v_1 - the nose.
- Each part has an ideal position p_i relative to nose.
  - Let $T_{1i}(l_1) = l_1 + p_i$,



$$E(l_1, \ldots, l_n) = \sum_{i=1}^{n} m_i(l_i) + \sum_{i=2}^{n} ||l_i - T_{1i}(l_1)||^2$$

Efficient minimization

$$L^* = \operatorname*{argmin}_{L} \left(\sum_{i=1}^{n} m_i(l_i) + \sum_{i=2}^{n} ||l_i - T_{1i}(l_1)||^2 \right)$$

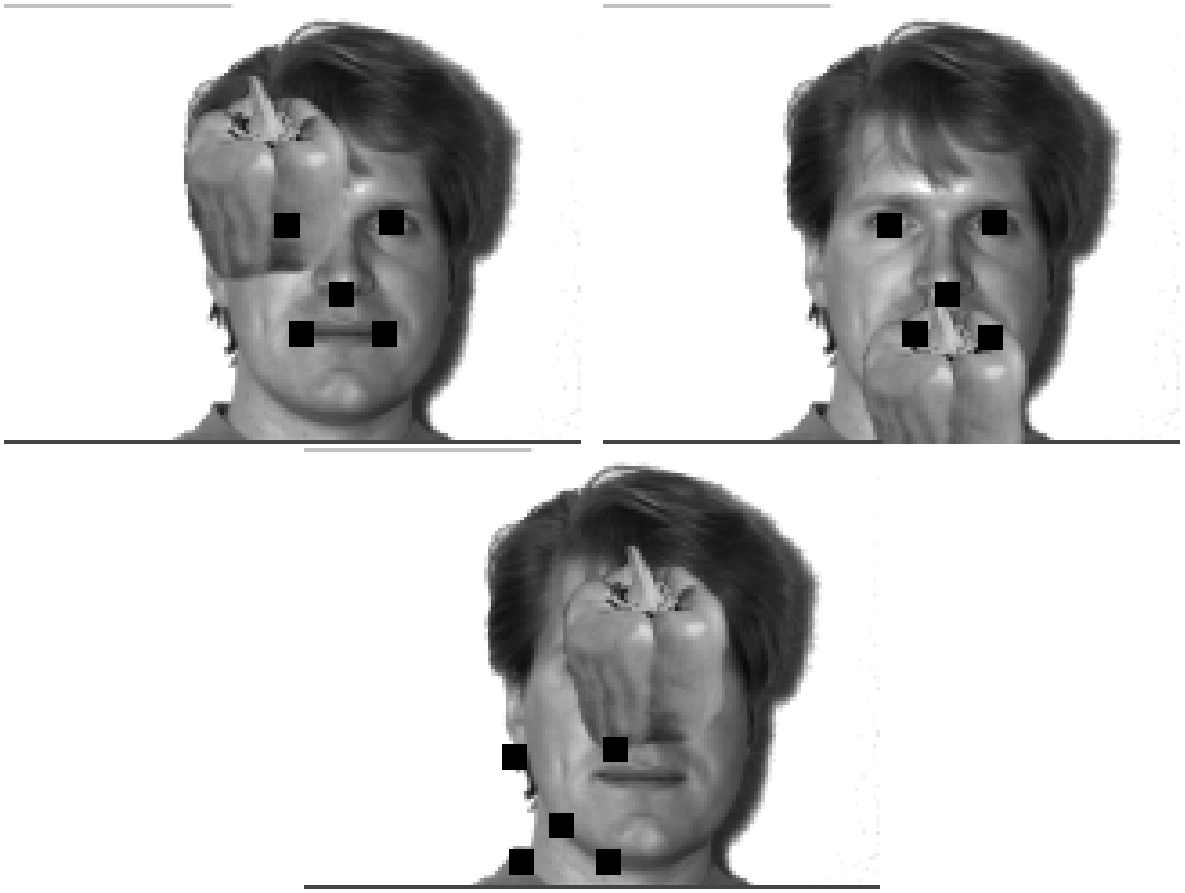
$$L^* = \operatorname*{argmin}_{L} \left(m_1(l_1) + \sum_{i=2}^{n} m_i(l_i) + ||l_i - T_{1i}(l_1)||^2 \right)$$

$$l_1^* = \operatorname*{argmin}_{l_1} \left(m_1(l_1) + \sum_{i=2}^{n} \min_{l_i} (m_i(l_i) + ||l_i - T_{1i}(l_1)||^2) \right)$$

$$l_1^* = \operatorname*{argmin}_{l_1} \left(m_1(l_1) + \sum_{i=2}^{n} \mathcal{D}_{m_i}(T_{1i}(l_1)) \right)$$

Matching results

Matching results

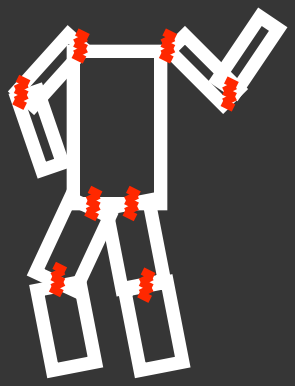


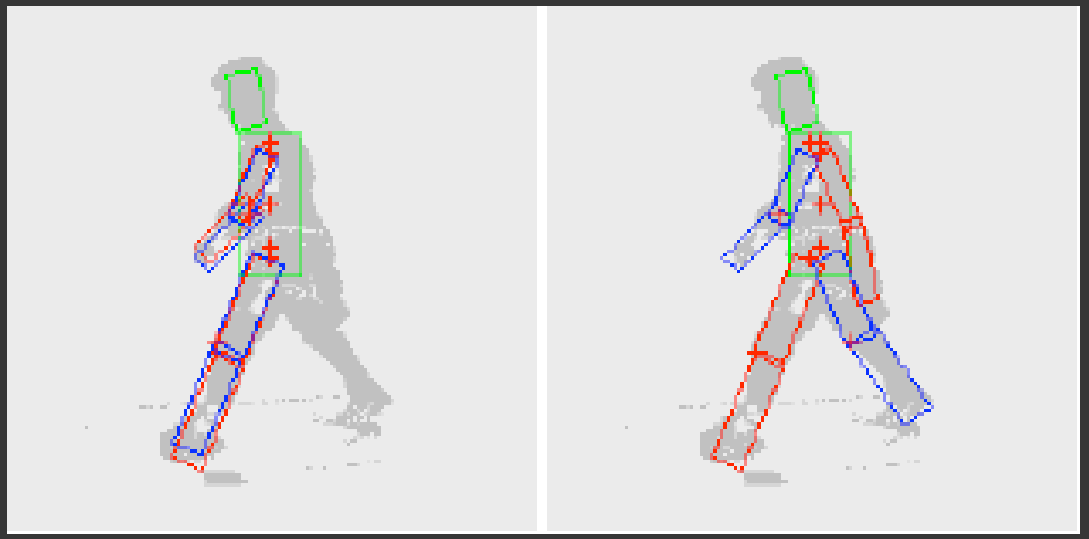
Summary

- Generic framework for part-based modeling.
- Global minimization for deformable objects can be fast.
- Soft detection avoids unnecessary early decisions.
- Partial occlusion is handled automatically.

Trouble with trees

- Limbs attracted to regions of high likelihood
  (local image evidence is double-counted)

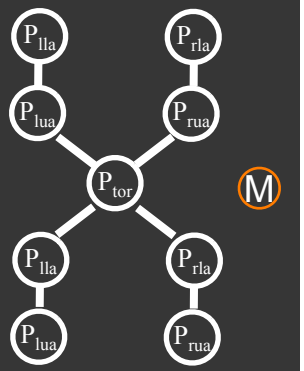


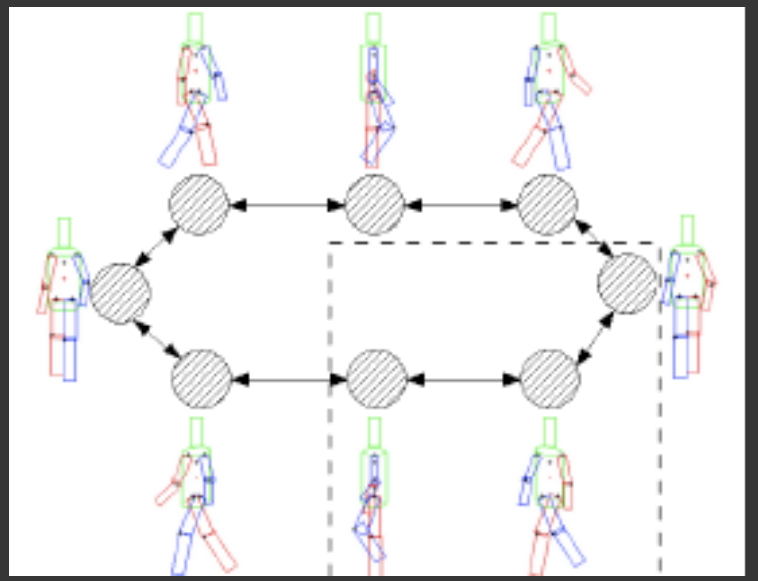


Lan & Huttenlocher, ICCV05

Tree extensions: augment model

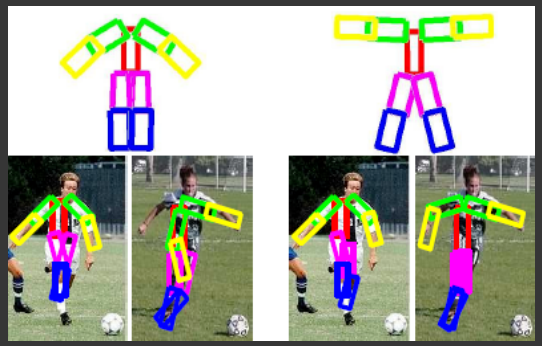
- Latent variable models (mixture component)





Lan & Huttenlocher, ICCV05
Lan & Huttenlocher, CVPR04

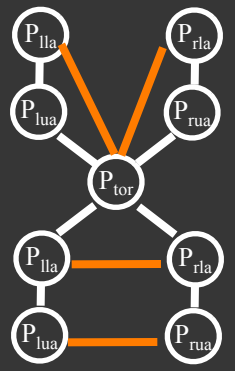
- Train tree model discriminatively (CRF)



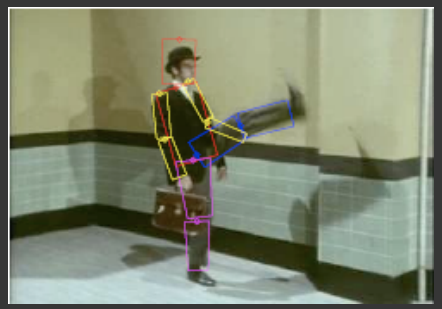
Ramanan & Sminchisescu CVPR06

Tree extensions: Don't use a tree!

Add loops enforcing non-occlusion

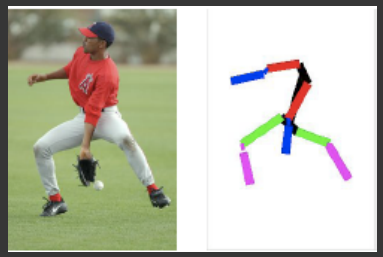


- Use loopy belief propogation (nonparametric msgs)



Sigal and Black CVPR06

- Combinatorial search (integer quadratic program)



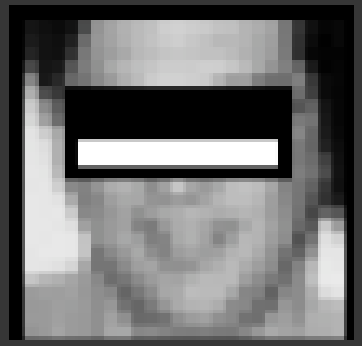
Mori et al, CVPR04

Ren et al, ICCV05

Part-based: What are good parts?

Build a part detector

- Face detector (adaboost, SVMs, NN)



Viola & Jones, IJCV01

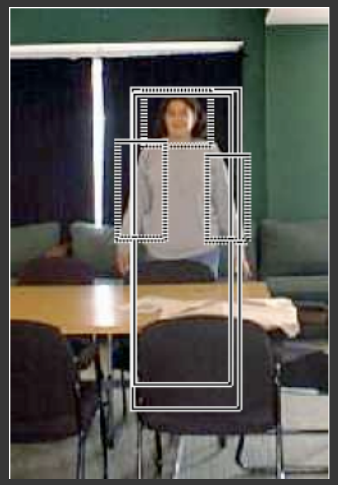
Schneiderman and Kanade, CVPR98

….

- Rich body of literature

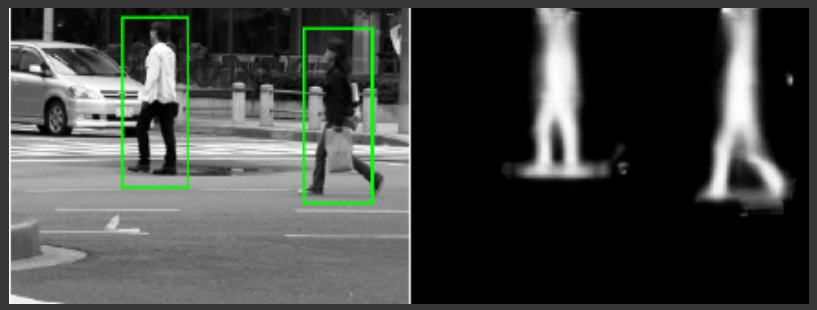
Part-based: What are good parts?

- Train a body part detector with SVM, adaboost, etc.



Mohan et al, PAMI01
Ronfard et al, ECCV02
Mikolajczyk et al ECCV04

- Learn a body part model
  -grayscale patches, filter responses



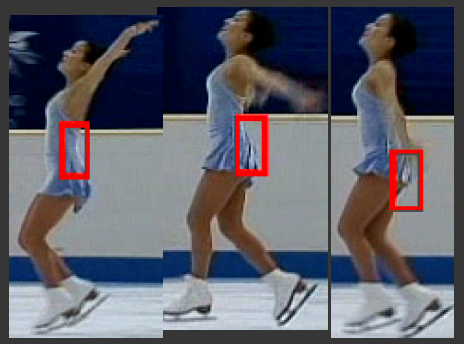
Liebe et al CVPR05
Roth et al CVPR04

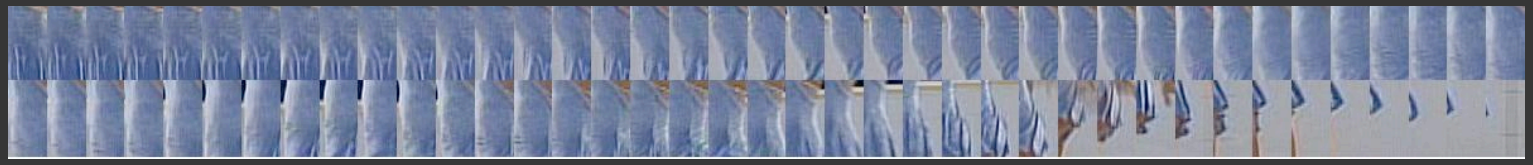
- Look for limb-like segments

Mori et al, CVPR04
Ren et al, ICCV05
Mori ICCV05

Try updating person-specific appearance

After hand-initializing…

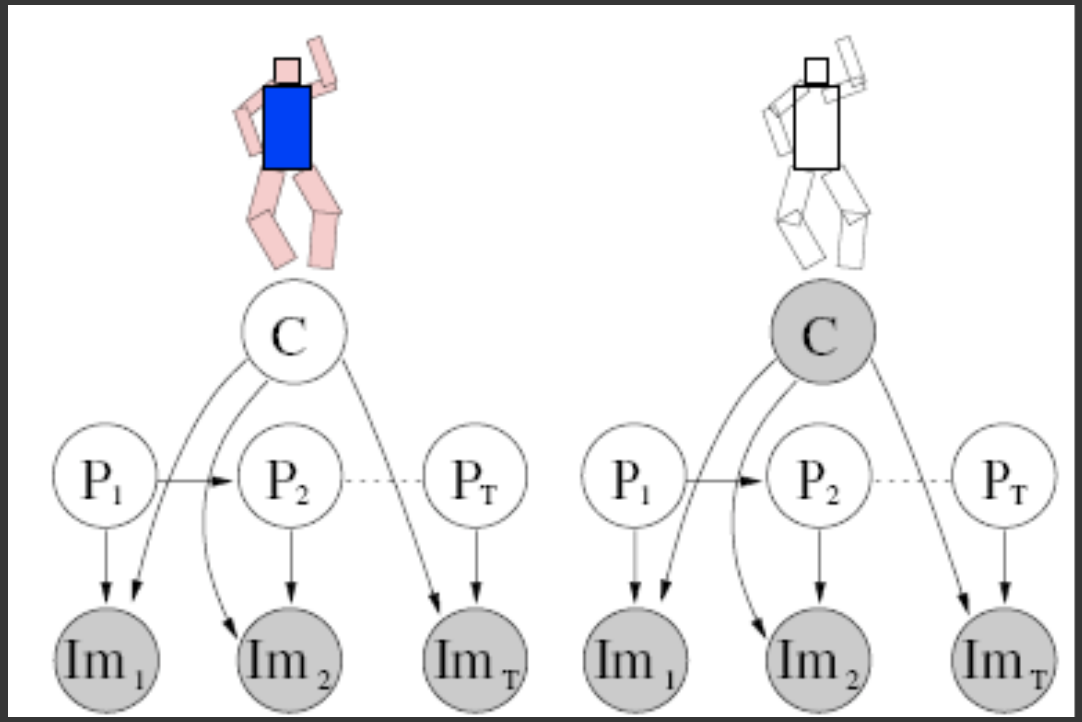




"Telephone Game" can causes drift

Can fix with a accurate data association (i.e. bg subtraction)

Model-based Tracking



If we know model *a priori* => regular Markov model

But model must necessarily be detuned

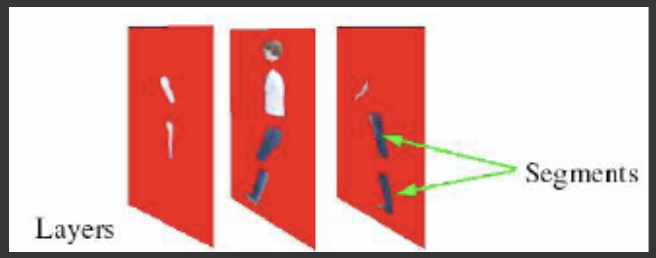
We want to learn template on-the-fly

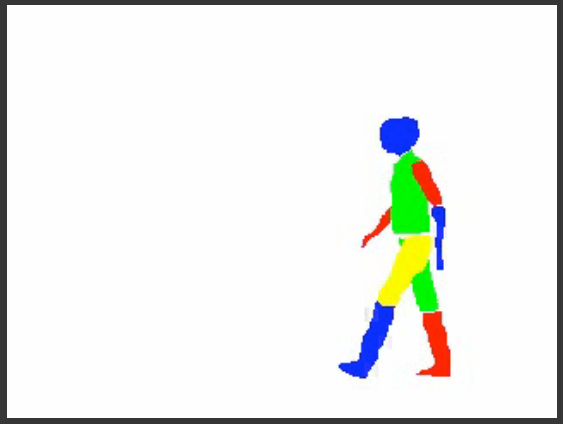
Building models on-the-fly by EM

Jojic & Frey, CVPR01

Learn templates, alpha masks, and depth ordering

Building models by EM (cont'd)



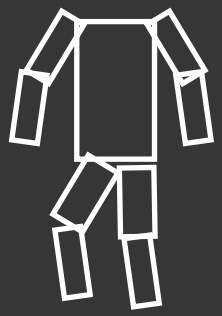


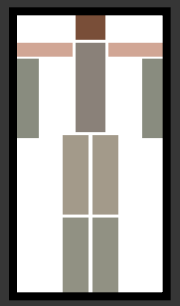
Kumar et al, ICCV05

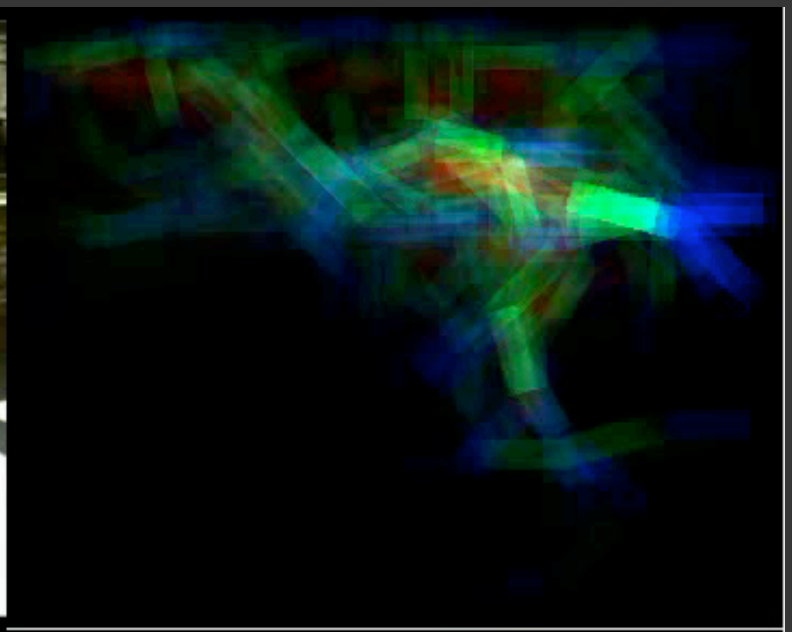
Add low-level segmentation cue to the model

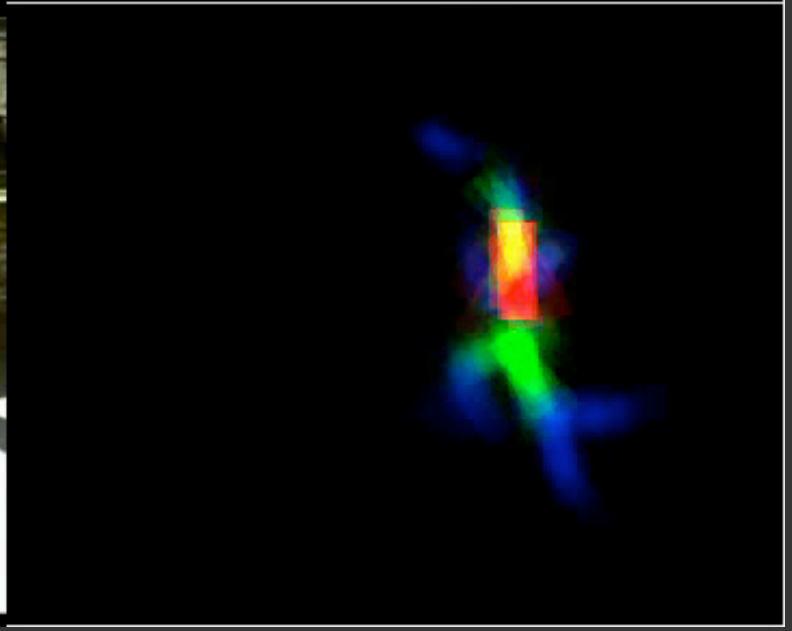
Add temporal illumination variable

Are learned models better?

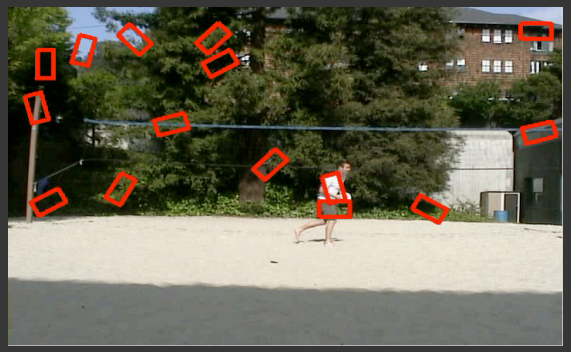




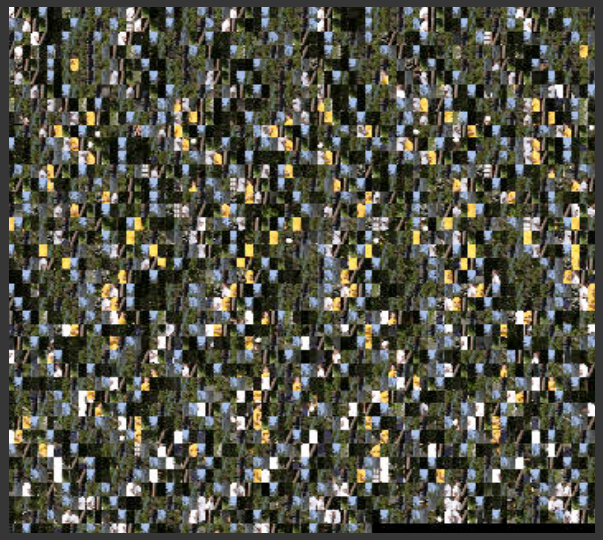




Build models by clustering candidate parts



detected torsos

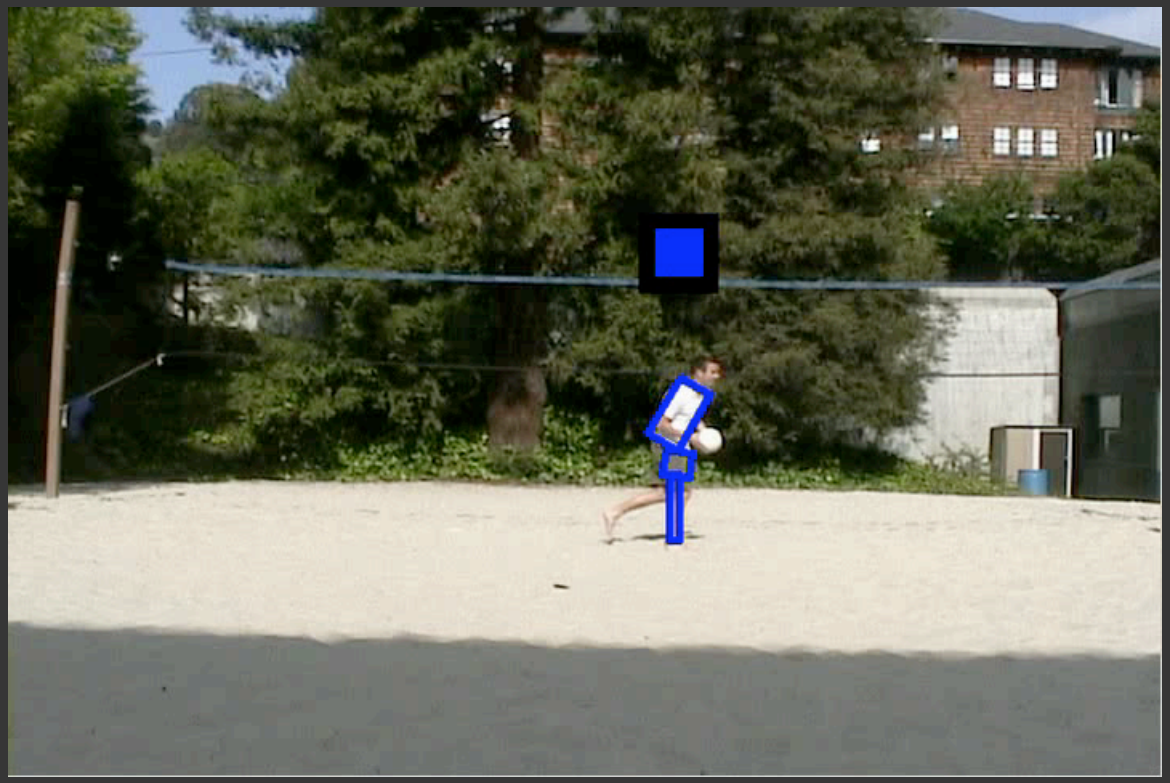


bag of detected torso patches

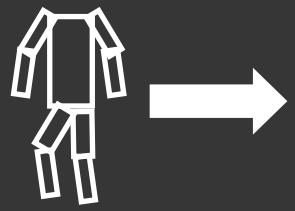
clustered detections

keep ones that don't move

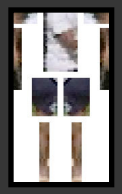
Track multiple people by model-building + detection



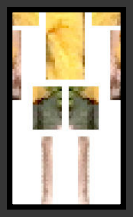
Ramanan & Forsyth CVPR03



person detector



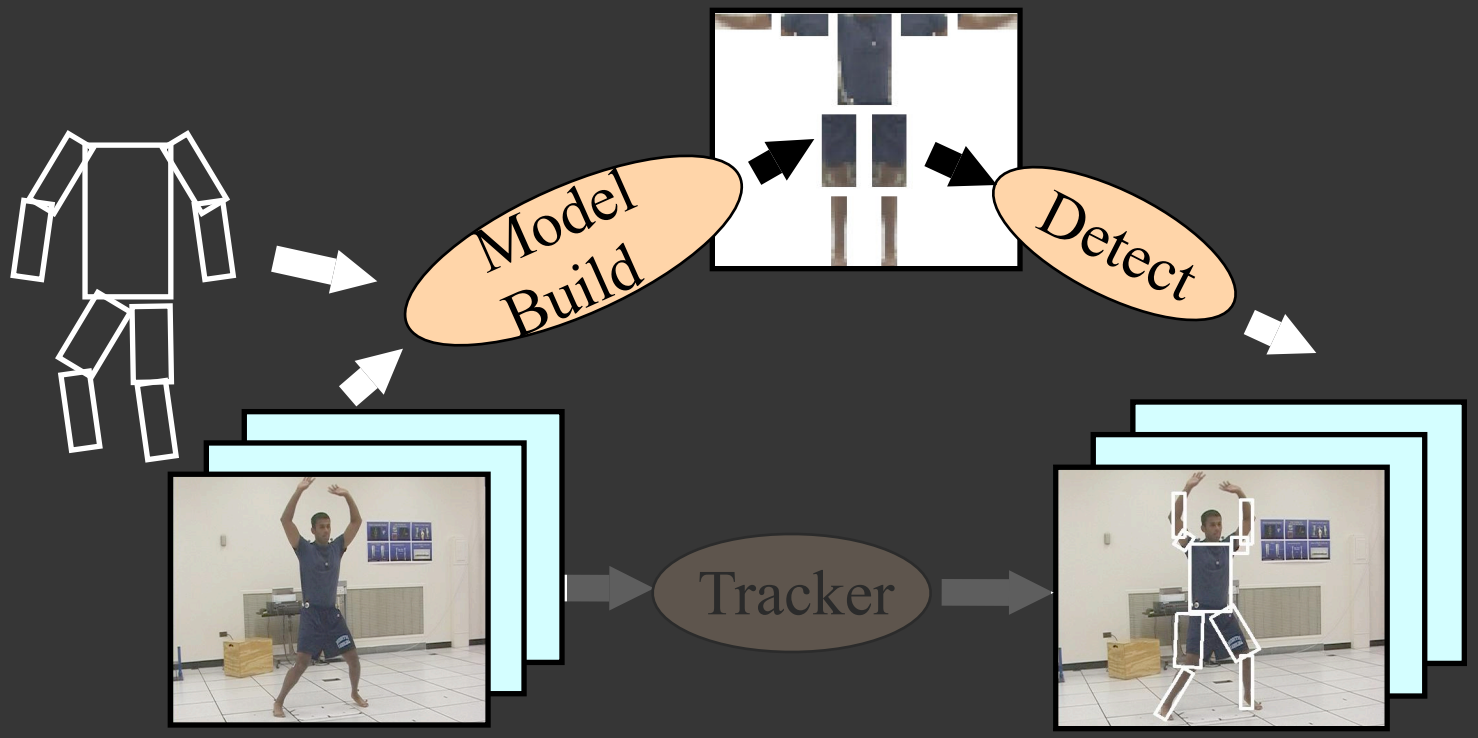
Deva detector



Bryan detector

John detector

How can we build models reliably?



Look for easy frames!

Which frames are easy?

People take on a variety of poses, aspects, scales

self-occlusion

rare pose

motion blur

non-distinctive pose

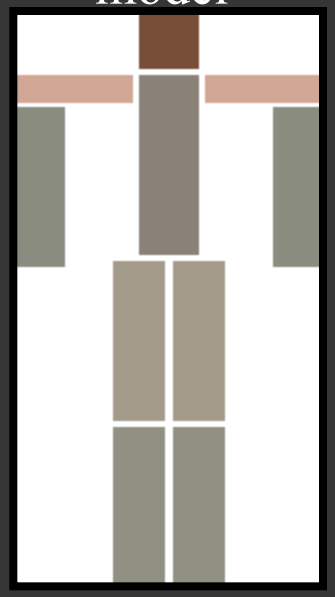
too small

just right
detect this

(Pick you're favorite)
window-based pedestrian classifier
XYT template
top-down exemplars

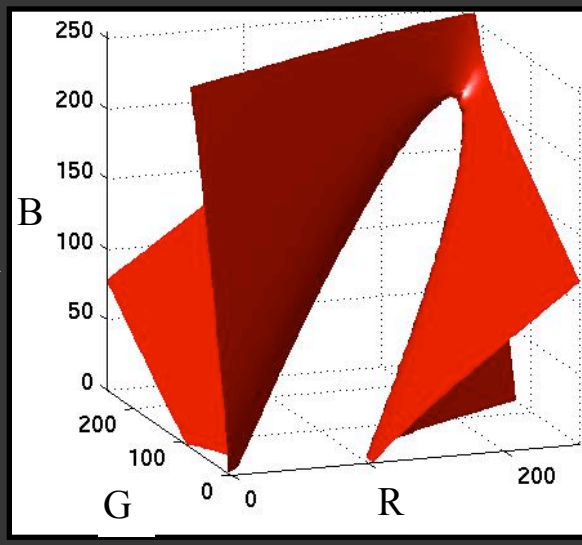
Build model

build model



limb pixel masks

learn limb classifier



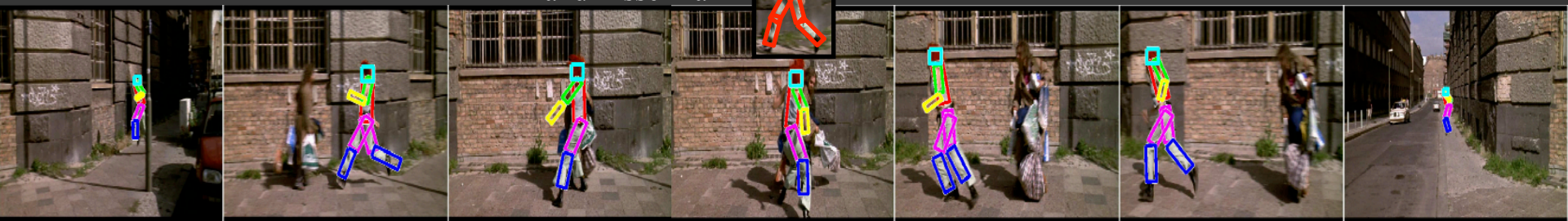
Sequence-specific
discriminative features for tracking

Collins et al. CVPR03
Avidan CVPR05
Ramanan et al CVPR05

`Run Lola Run'

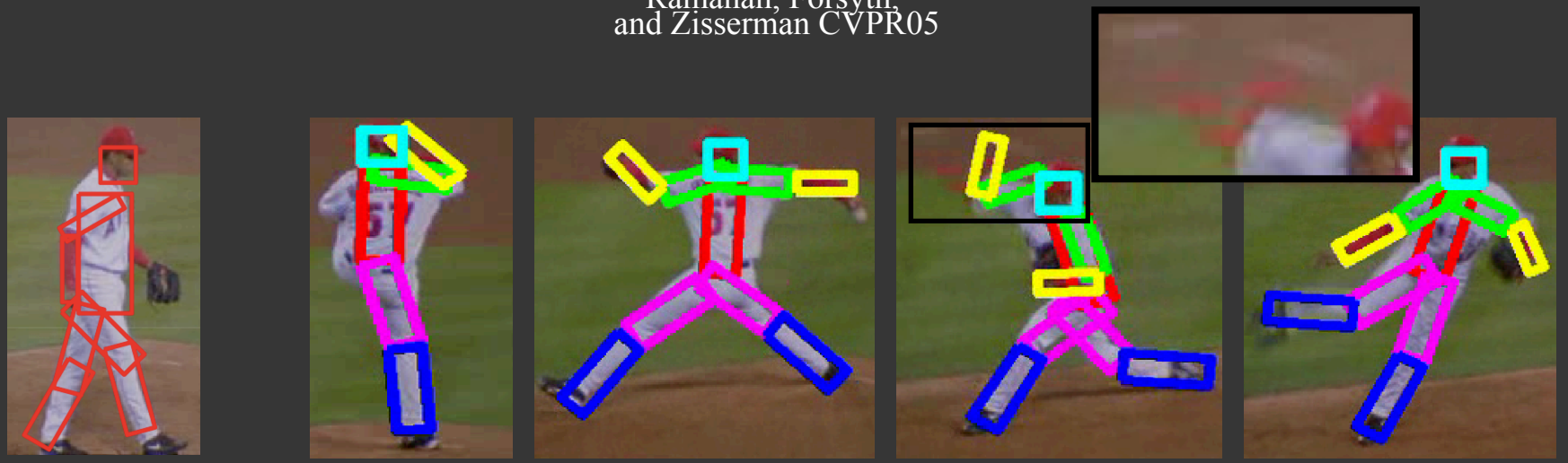
Ramanan, Forsyth, and Zisserman

CVPR05



How likely is a ‘typical’ pose?

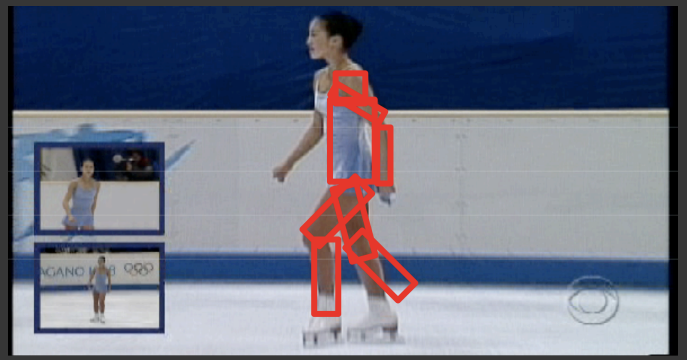
Ramanan, Forsyth, and Zisserman CVPR05



motion blur &interlacing

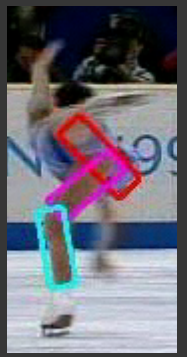
Track long footage (7600 frames)

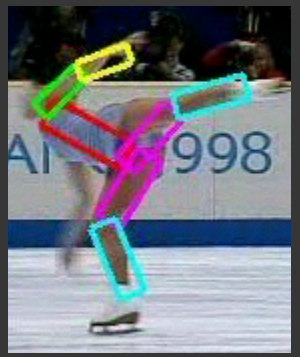
Ramanan, Forsyth, and Zisserman CVPR05



extreme pose

motion blur





fast movement

Pictorial Structure Revisited: Model Components [Schiele cvpr09]

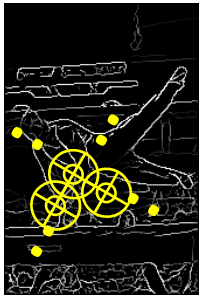
Appearance Model:

Prior and Inference:

Local Features

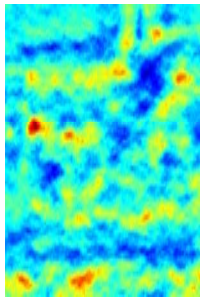
orientation 1

orientation K

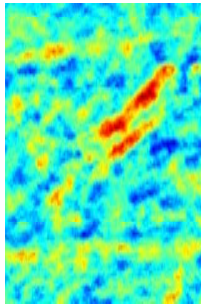


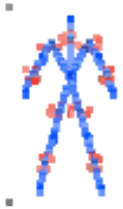
AdaBoost

likelihood of part 1

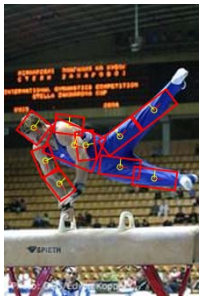


likelihood of part N





estimated pose



part posteriors

Detection and Segmentation with Poselets

Lubomir Bourdev

Subhransu Maji

Jitendra Malik

EECS U.C. Berkeley

This presentation is based on the paper:

Poselets: Body-Part Detectors Trained Using 3D Human Pose Annotations

Lubomir Bourdev and Jitendra Malik

ICCV 2009

Agenda

- What is a Poselet
- How do we construct poselets?
- How do we use poselets for:
  - Detection
  - 3D Pose localization
  - Segmentation
- Results on VOC
- Conclusion

Agenda

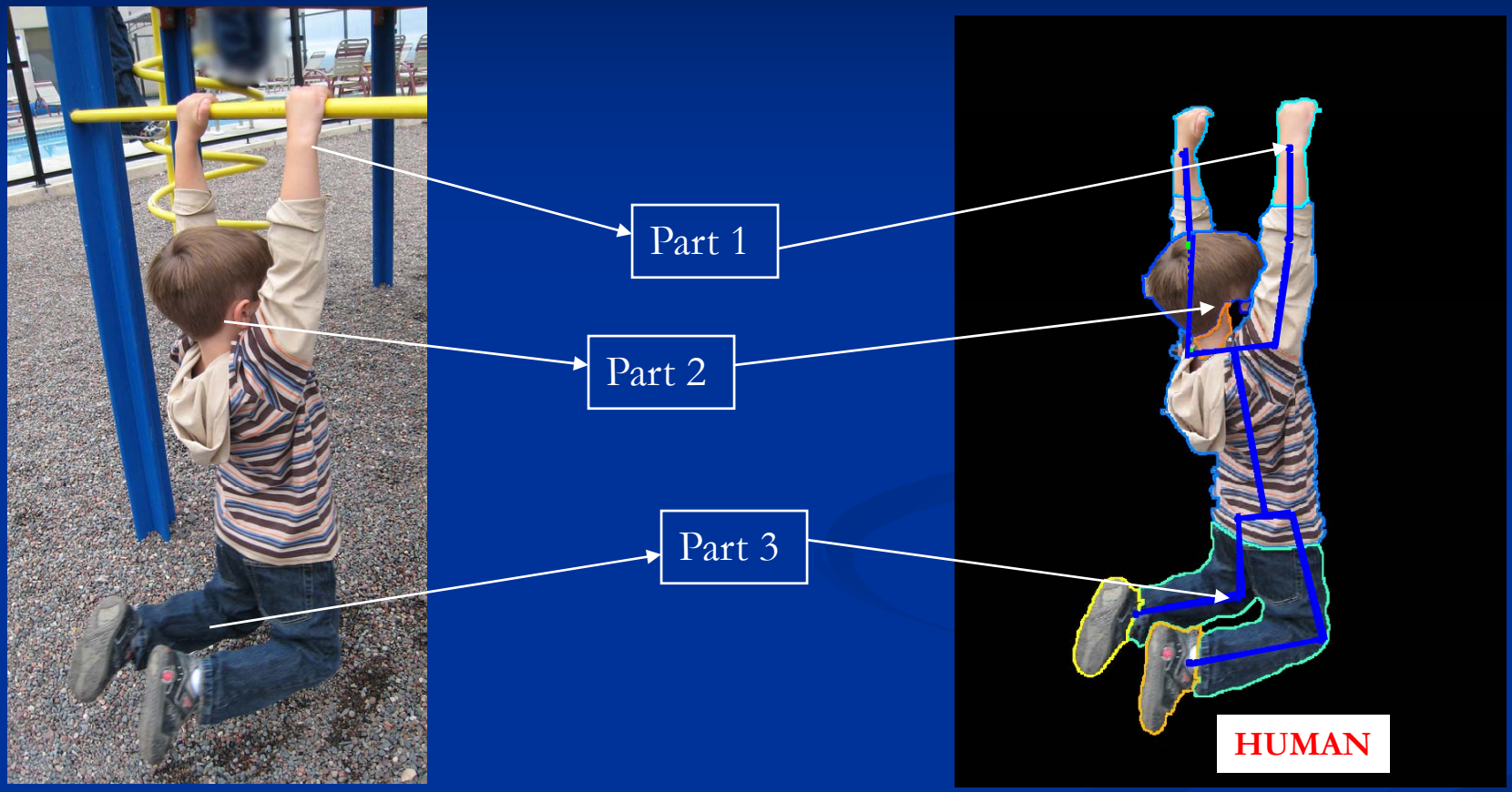
- What is a Poselet
- How do we construct poselets?
- How do we use poselets for:
  - Detection
  - 3D Pose localization
  - Segmentation
- Results on VOC
- Conclusion

Goal

Detection, localization, 3D Pose reconstruction, and segmentation of people

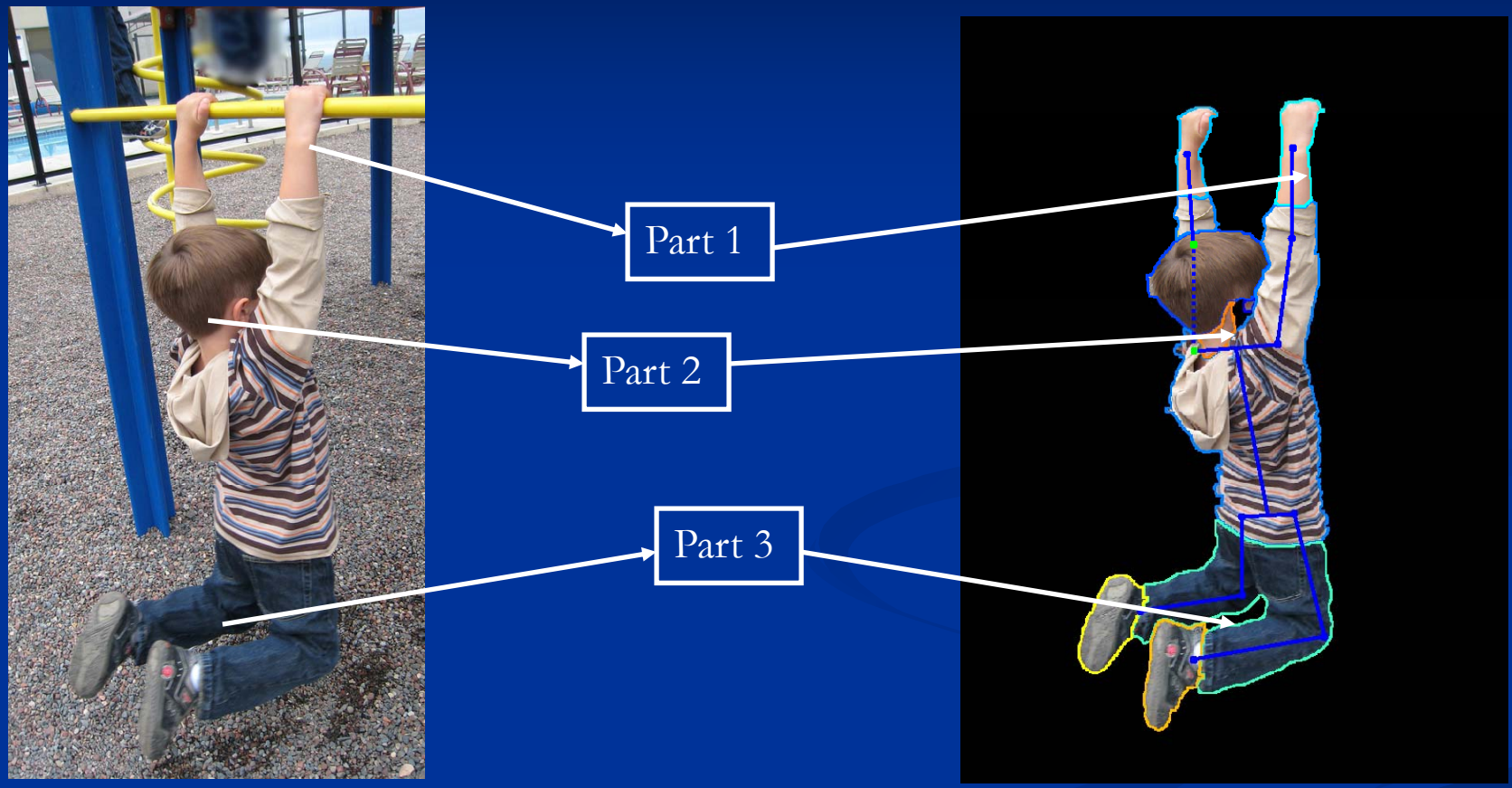
Human Detection is Challenging

Solution: Part-Based Detectors

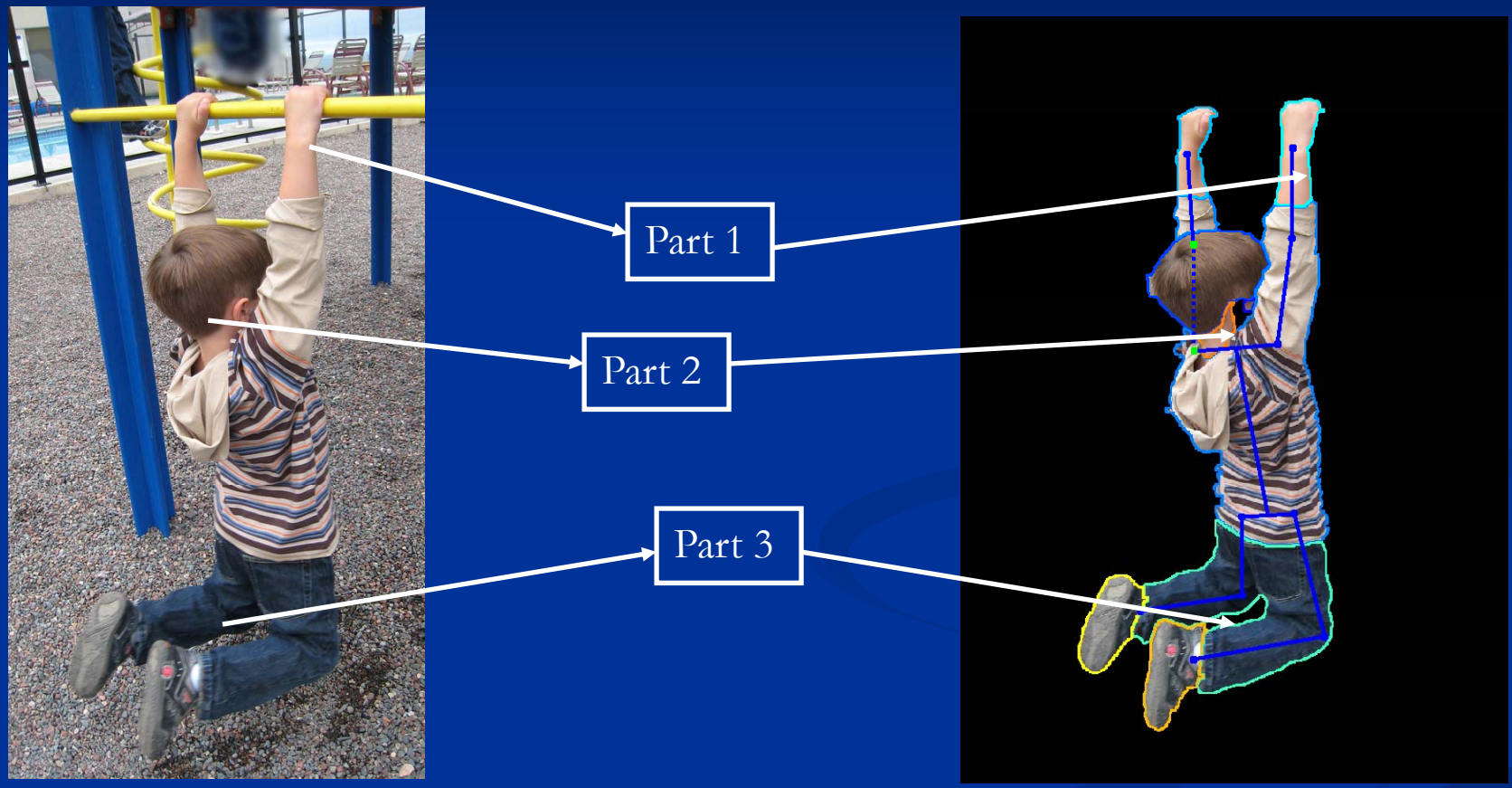


But how do we select good parts?

Properties of good parts



Properties of good parts

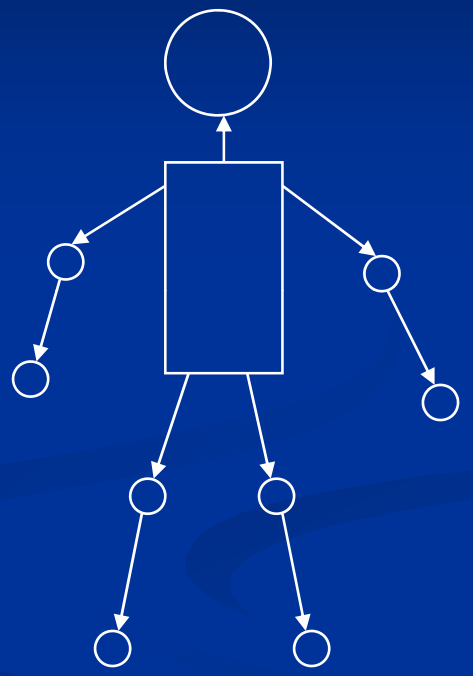


1. Parts must be tightly clustered in appearance
2. Parts must be tightly clustered in 3D pose space

Prior Work: Hard-coding parts

Example: Pictorial Structures

- Body parts defined in explicit hard-coded tree structure
- Matches physical body constraints
- Efficient to evaluate



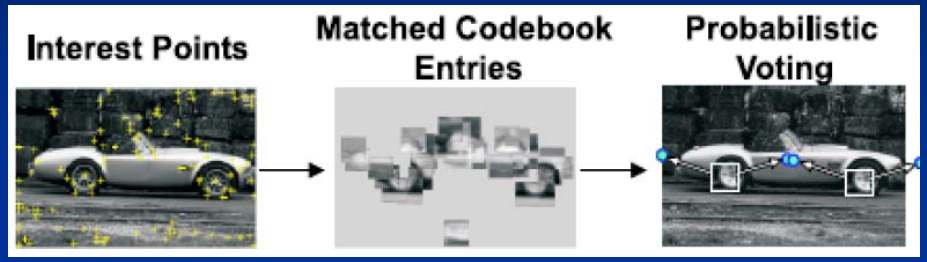
Pictorial Structures

Tight in configuration space

Not tight in visual space

[Felzenszwalb & Huttenlocher IJCV'05]
[Ramanan NIPS'06]
[Ferrari et al. CVPR'08]

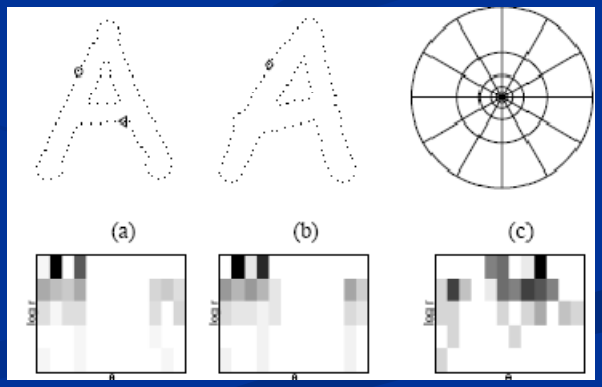
Prior Work: Learning Parts



Implicit Shape Models [Leibe et al. ECCV 2004]

Constellation Model
[Fergus et al. CVPR'03]

- Body parts emerge from training
- Parts chosen based on appearance



Shape Context (for people)
[Mori & Malik. ECCV'02]

Tight in visual space

Not tight in configuration space

Some Prior Research

Parts clustered in configuration space

Generalized Cylinders
[Novatia, Binford, AI77]

Pictorial Structures
[Felzenswalb, Huttenlocher IJCV05]
[Ramanan NIPS 06]
[Andriluka, Roth, Schiele CVPR 09]

Parts clustered in appearance space

Holistic Methods (pedestrians)
[Dalal, Triggs, CVPR05]

Learning Parts from the Image
[Leibe et al ECCV04]
[Fergus et al, CVPR 03]
[Mori, Malik, ECCV02]

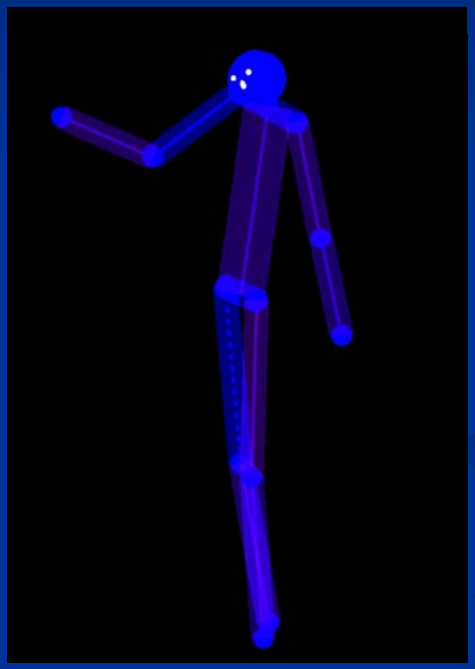
Our approach combines the strengths of both prior research directions
This requires extra annotations

Our Approach

- We combine the benefits of both prior research directions
- Our parts (called Poselets) are tight in both visual and configuration space
- We leverage a novel human annotation data set H3D

H3D The Human 3D Dataset

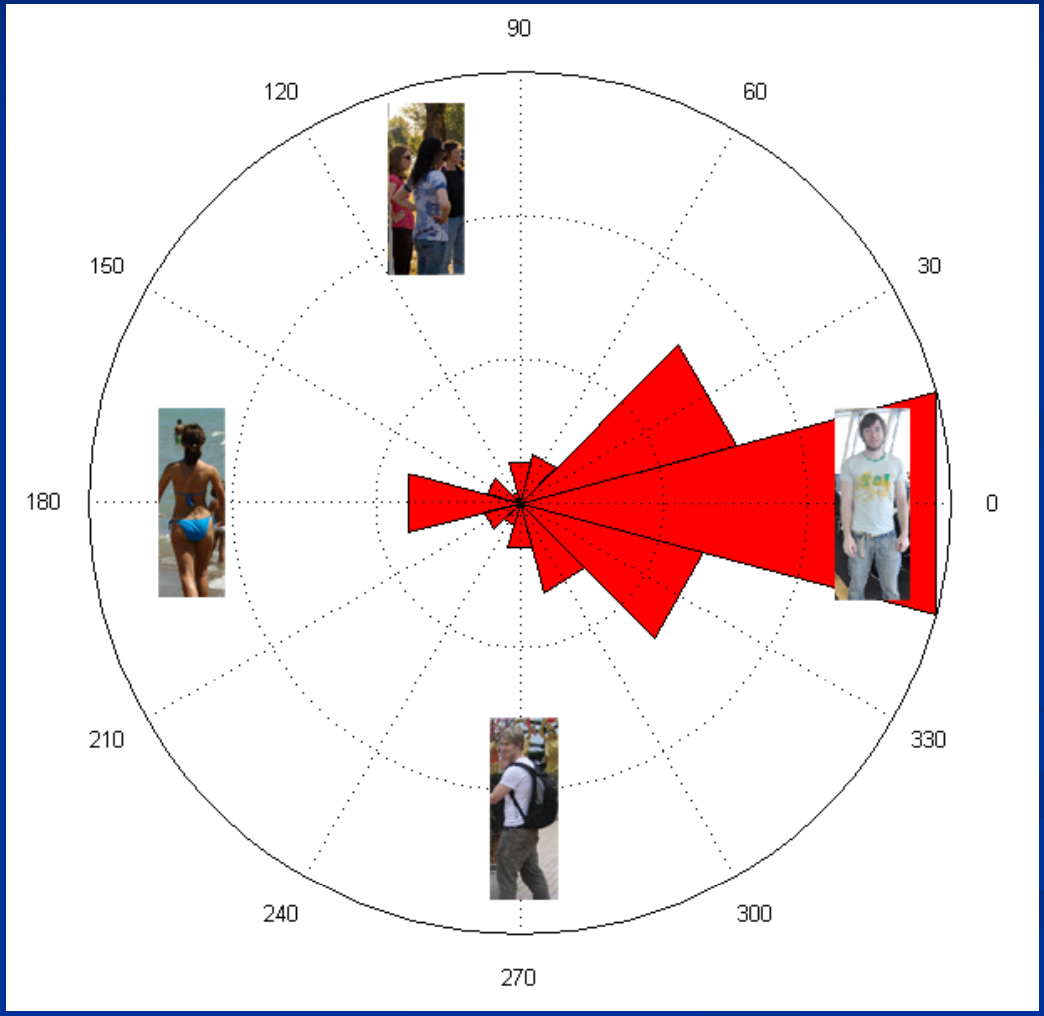
Annotated keypoints



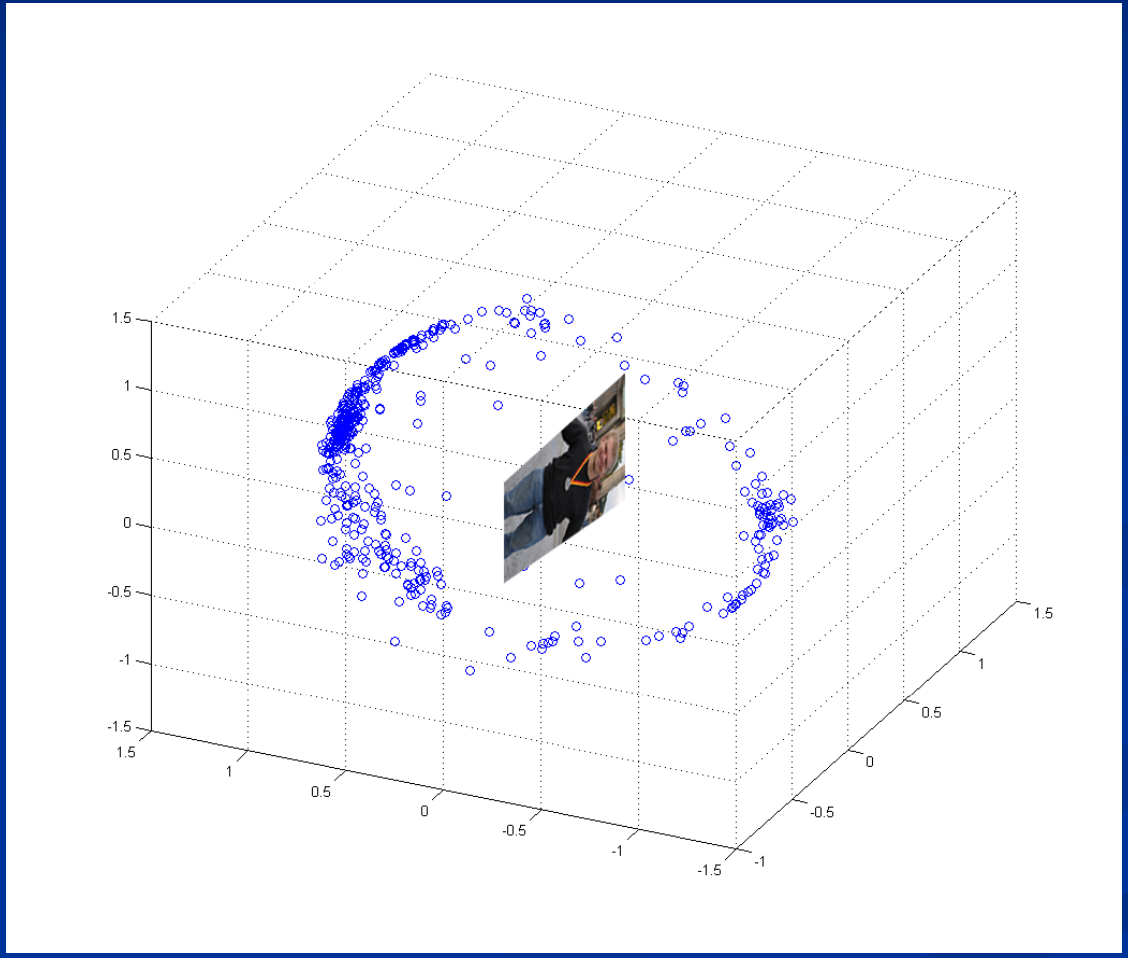
3D Pose

Segmented regions

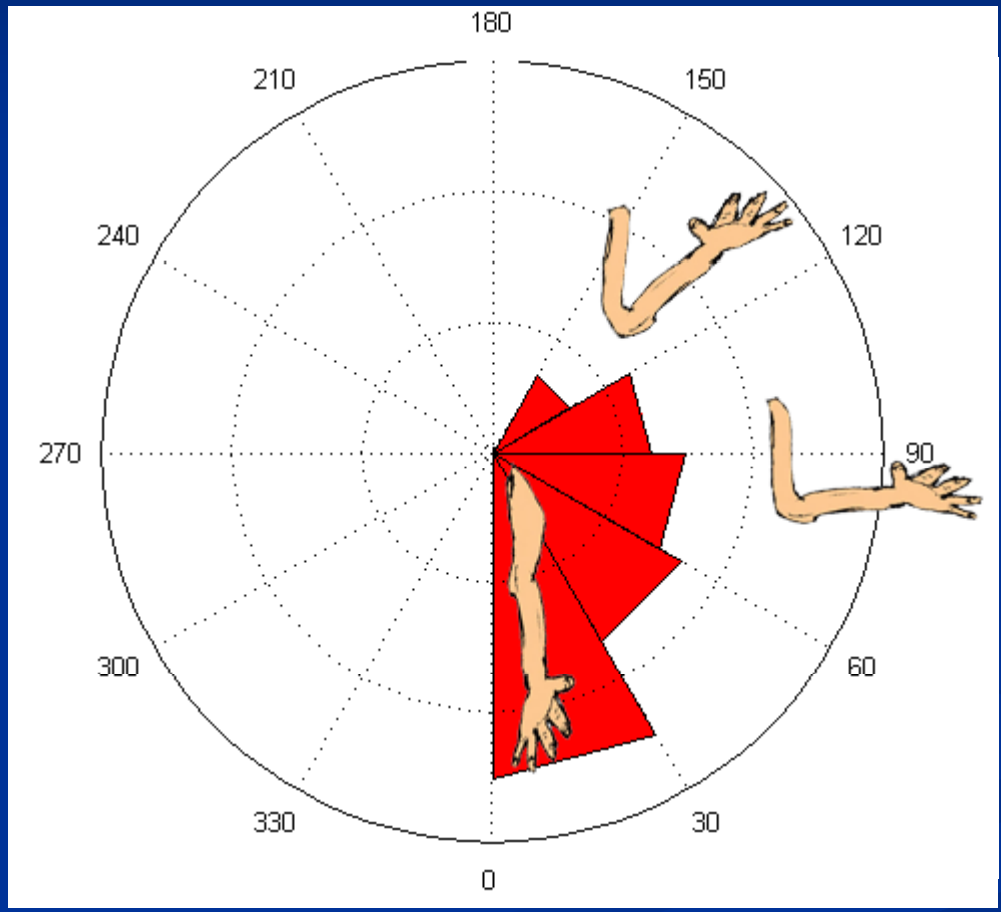
Expected Camera Azimuth



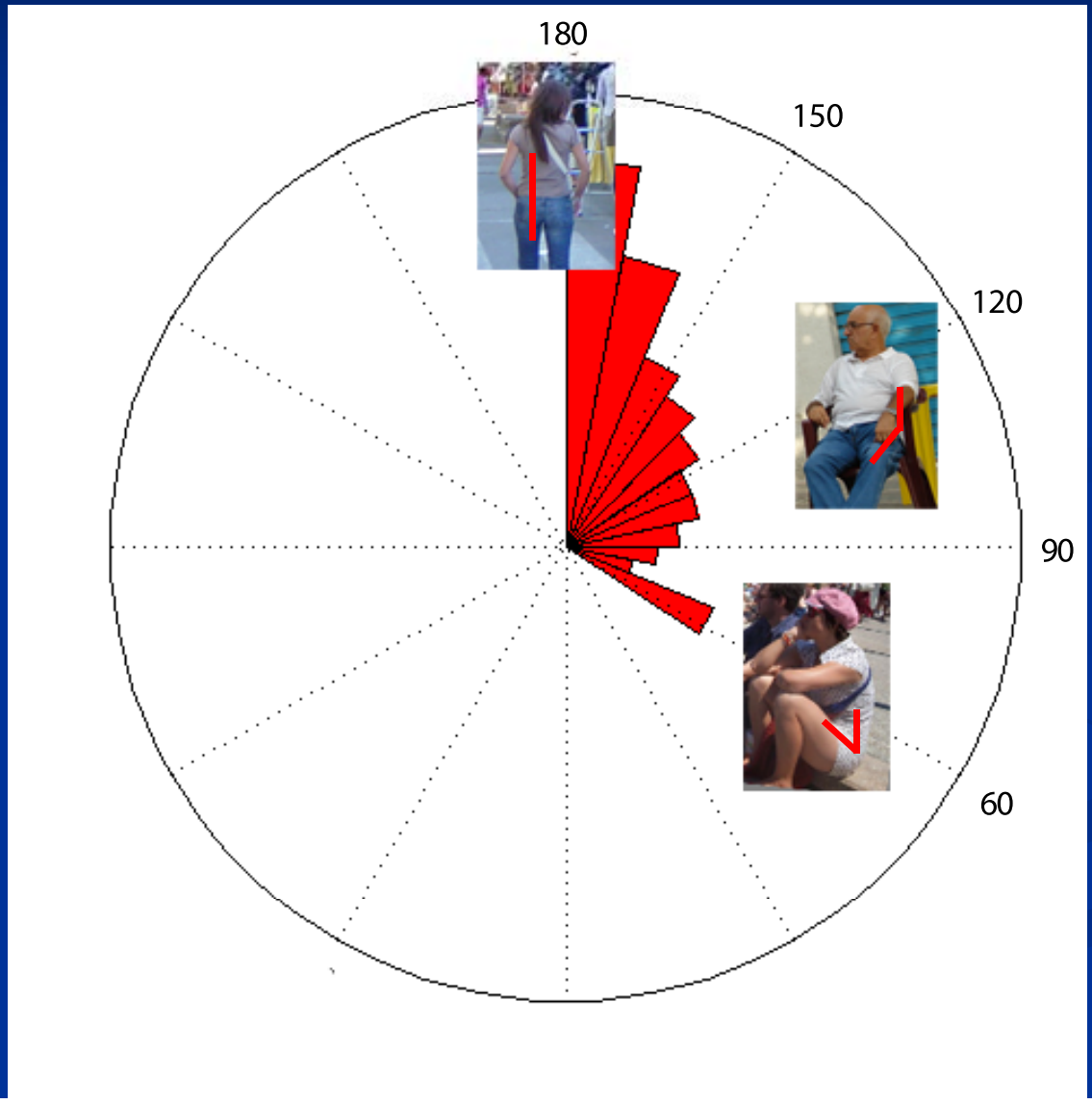
Expected Azimuth + Elevation



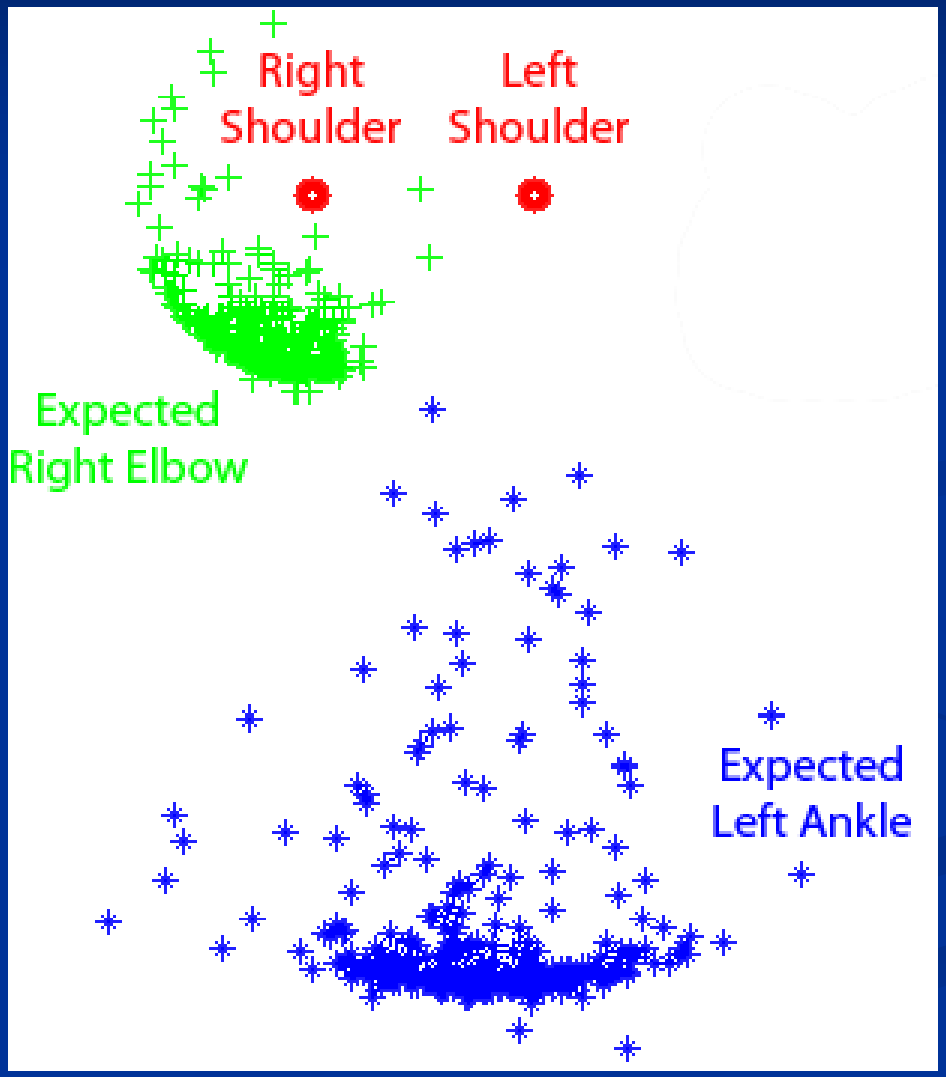
Expected Bending Angle of Left Arm



Expected torso/leg angle



Expected Keypoint locations in 2D



Expected Pixel Labels

Sitting people with background removed

H3D Dataset Detail

- 1000 full 3D annotations
- Mirrored to 2000 (1500 training, 500 test set)
- Annotation tool on my web page in Java3D
- Flickr Creative Commons Attribution license
- H3D Sponsored by Hewlett Packard

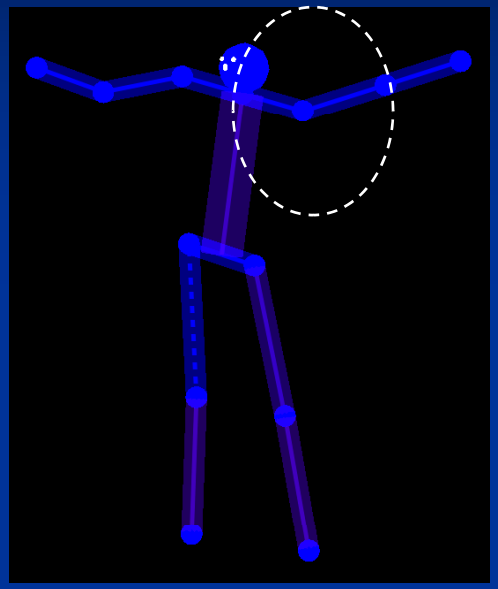
Agenda

- What is a Poselet
- How do we construct poselets?
- How do we use poselets for:
  - Detection
  - 3D Pose localization
  - Segmentation
- Results on VOC
- Conclusion

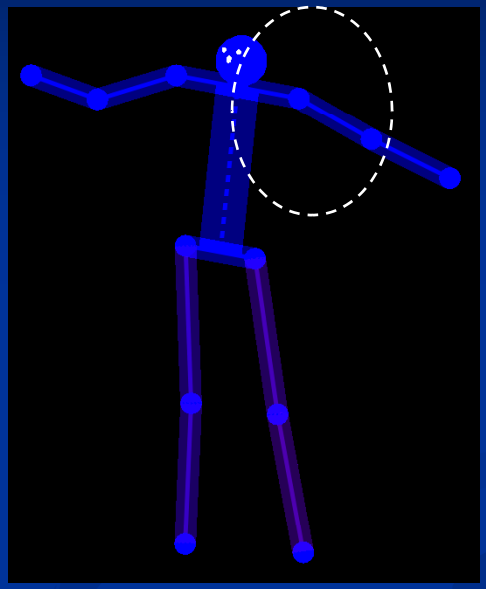
Examples of poselets

Patches are often far **visually**, but they are close **semantically**

Distance in Configuration Space



s



r

Sum of squared errors modulated by a spatial Gaussian *w*:

$$d_s(r) = \sum_i w_s(i) ||\mathbf{x_s}(i) - \mathbf{x_r}(i)||_2^2$$

Matching Patches Across Images

Use weighted least squares to find the similarity transform that minimizes the distance.

$$X^T W X \hat{\beta} = X^T W y$$

Query

Match 1

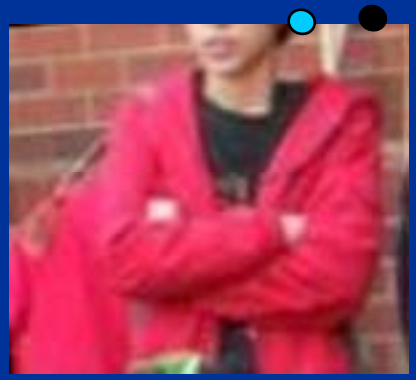
Match 2

Weaker Match

Matching Patches Across Images

Use weighted least squares to find the similarity transform that minimizes the distance

Query



Match 1

Match 2

Weaker Match

Matching Patches Across Images

Use weighted least squares to find the similarity transform that minimizes the distance:

$$X^T W X \begin{bmatrix} t_x \\ t_y \\ s \\ \theta \end{bmatrix} = X^T W y$$

Query

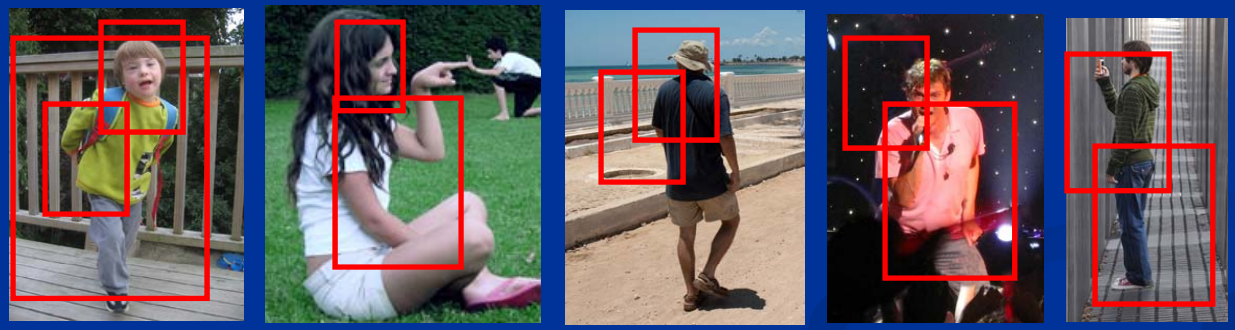
Match 1

Match 2

Weaker Match

How do we find poselets?

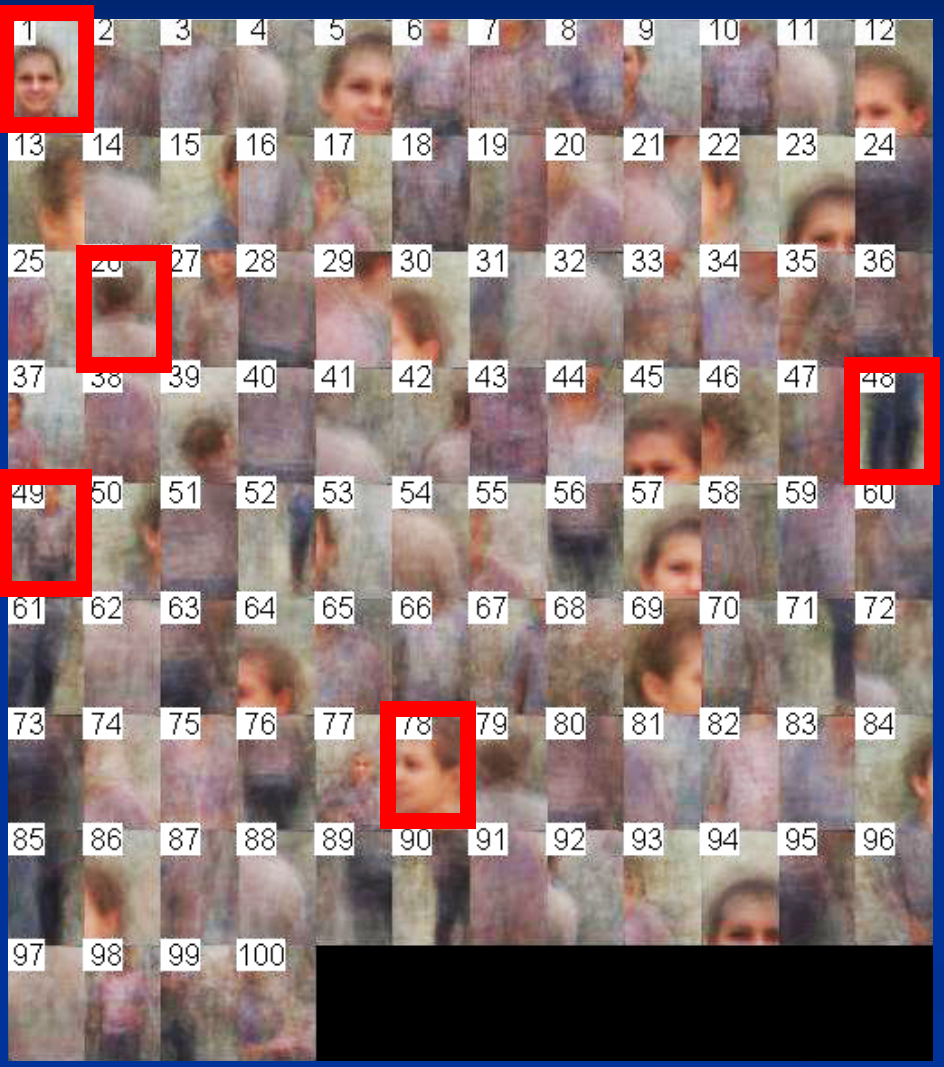
- Choose thousands of random windows, generate poselet candidates, train linear SVMs



- Filter out poselet candidates that don't train well or have few training examples

- Bootstrap the remaining poselets

Examples of Poselets



Agenda

- What is a Poselet
- How do we construct poselets?
- How do we use poselets for:
  - Detection
  - 3D Pose localization
  - Segmentation
- Results on VOC
- Conclusion

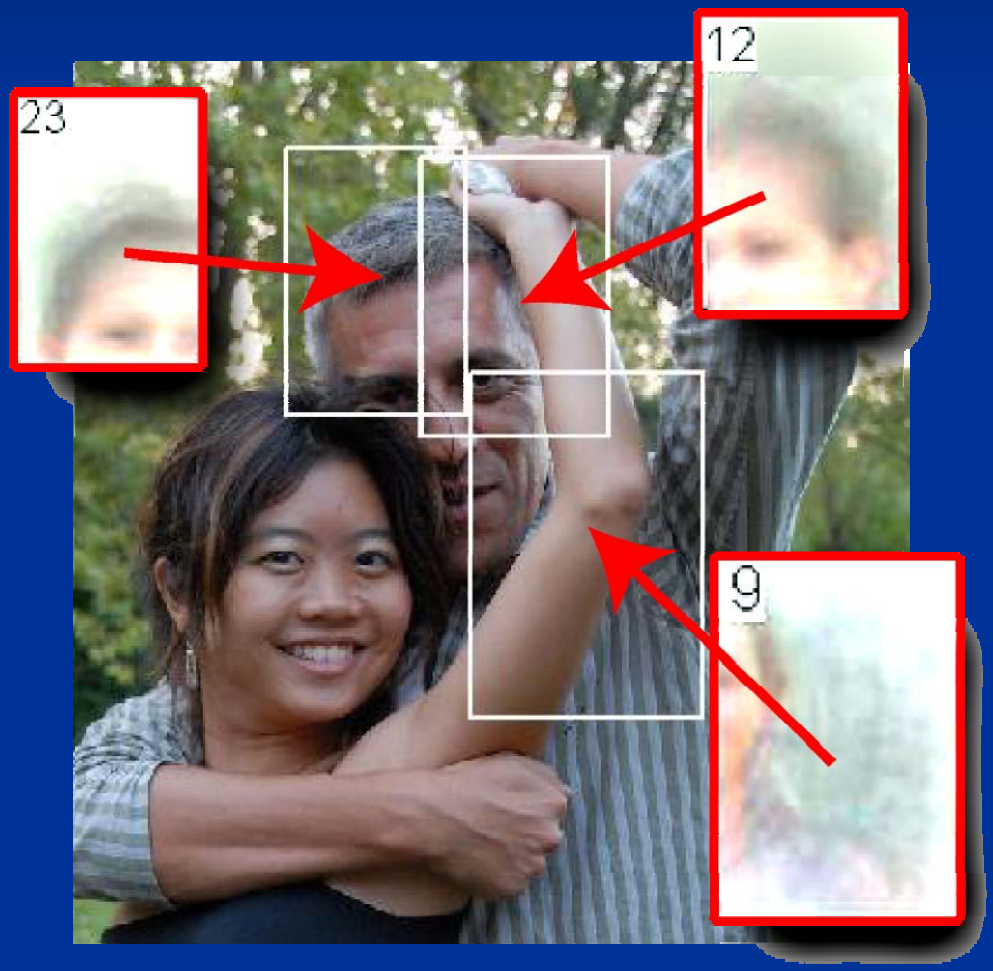
Fit torso location for each poselet

- Examples of a poselet:

- H3D supplies the torsos:

Expected Torso location

Torso Detector using Poselets



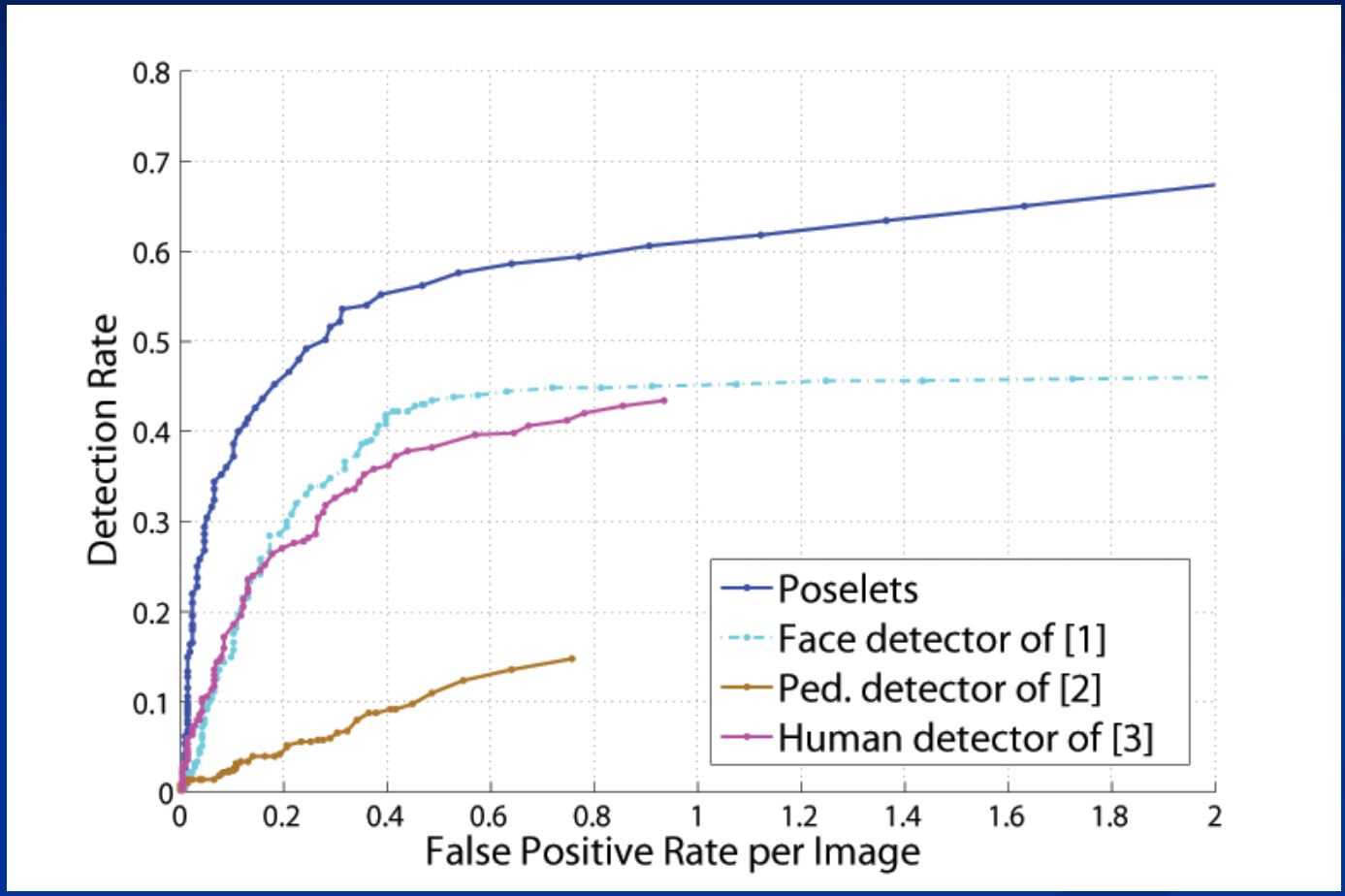
- Detect poselets
- Hough-vote for each torso location
- Score each cluster:

$$S(x) = \sum_i w_i a_i(x)$$

$a_i(x)$ { Score of poselet i at location x

w_i { Weight of poselet i learned via M^2HT [Maji/Malik CVPR09]

ROC on H3D Test Set

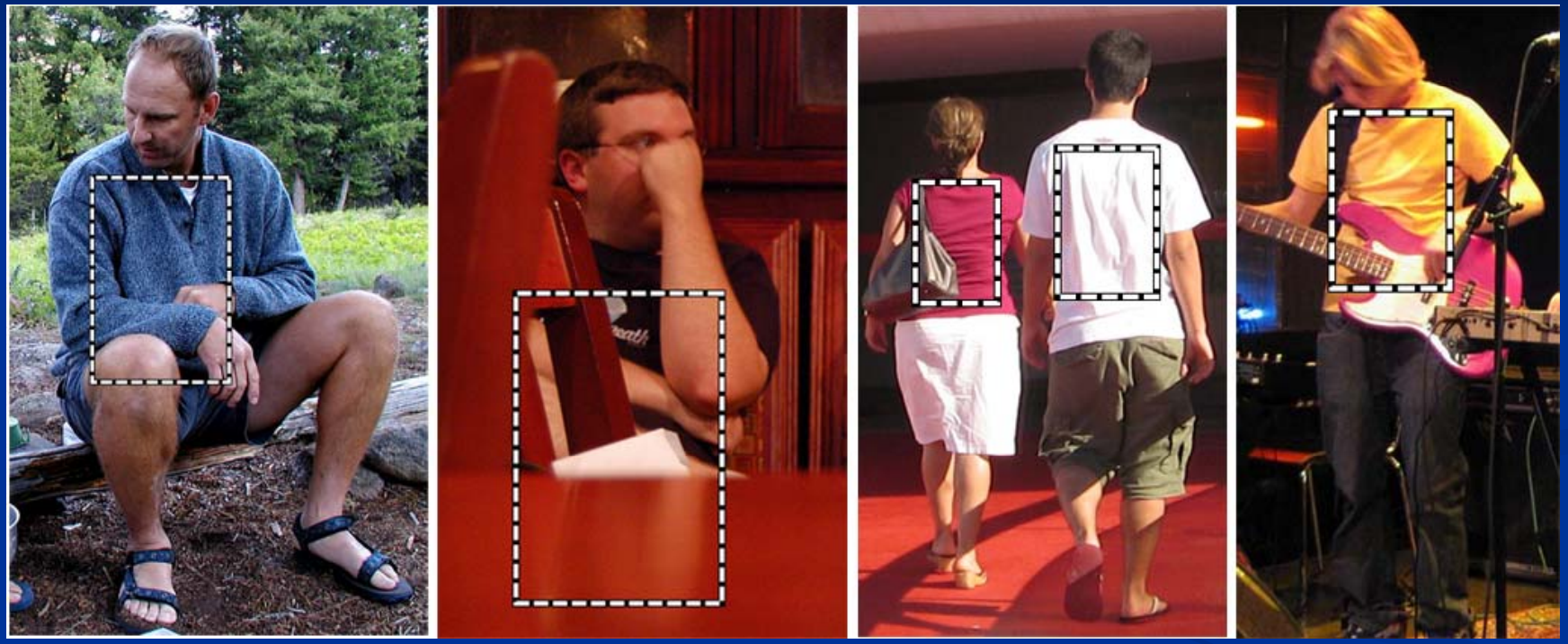


[1] Bourdev & Brandt CVPR 2005

[2] Dalal & Triggs CVPR 2005

[3] Felzenszwalb, Mcallester & Ramanan,CVPR 2008

Example Torso Hits

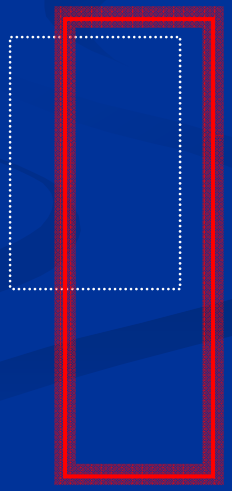


To predict object bounds, replace torso with bounds

- Examples of a poselet:

- H3D supplies the bounds:

Expected Object Bounds



Agenda

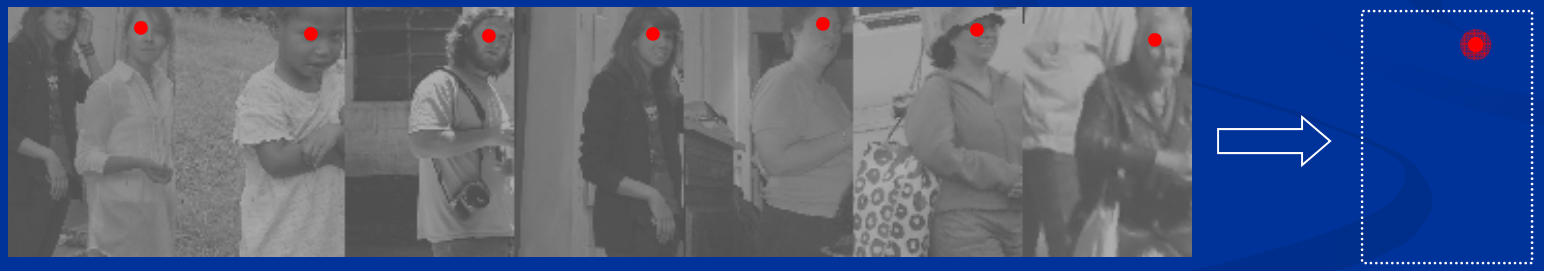
- What is a Poselet
- How do we construct poselets?
- How do we use poselets for:
  - Detection
  - 3D Pose localization
  - Segmentation
- Results on VOC
- Conclusion

Fit Keypoint Location from a Poselet

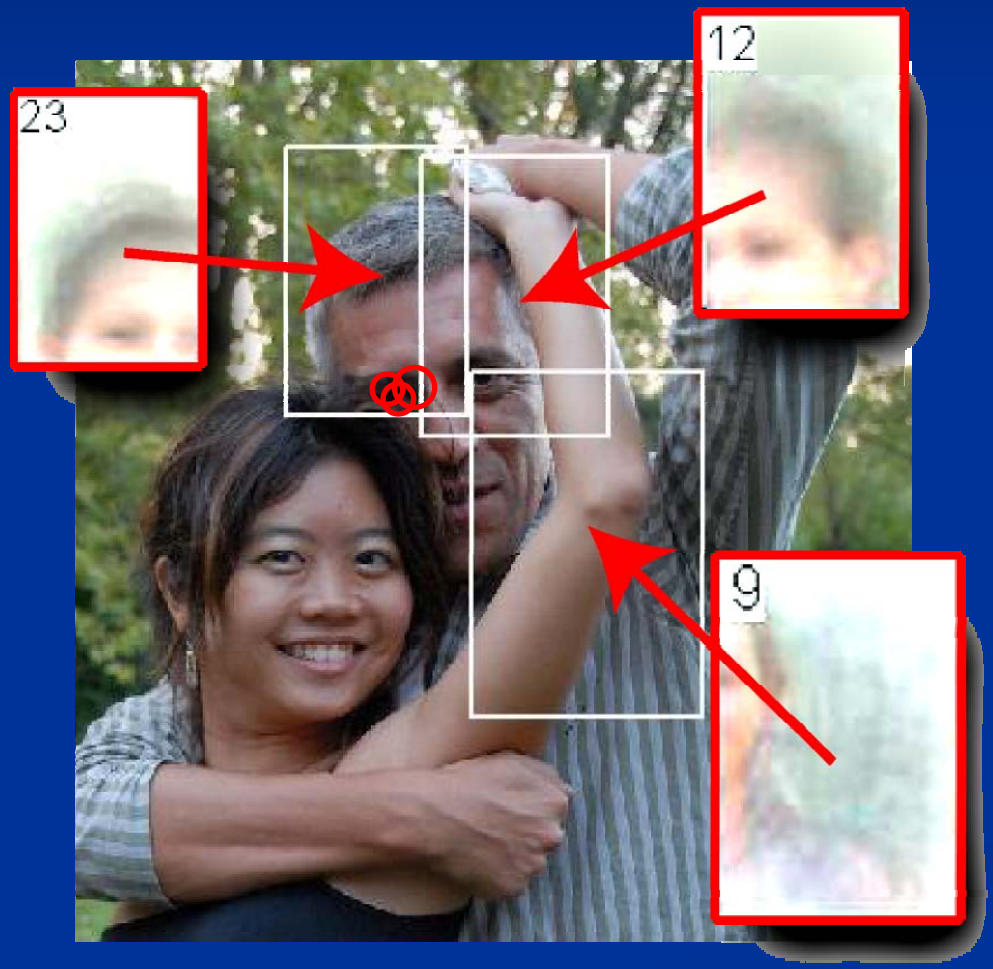
- Examples of a poselet:

- H3D supplies right eye locations:

Expected Right Eye Location



Keypoint Detection using Poselets

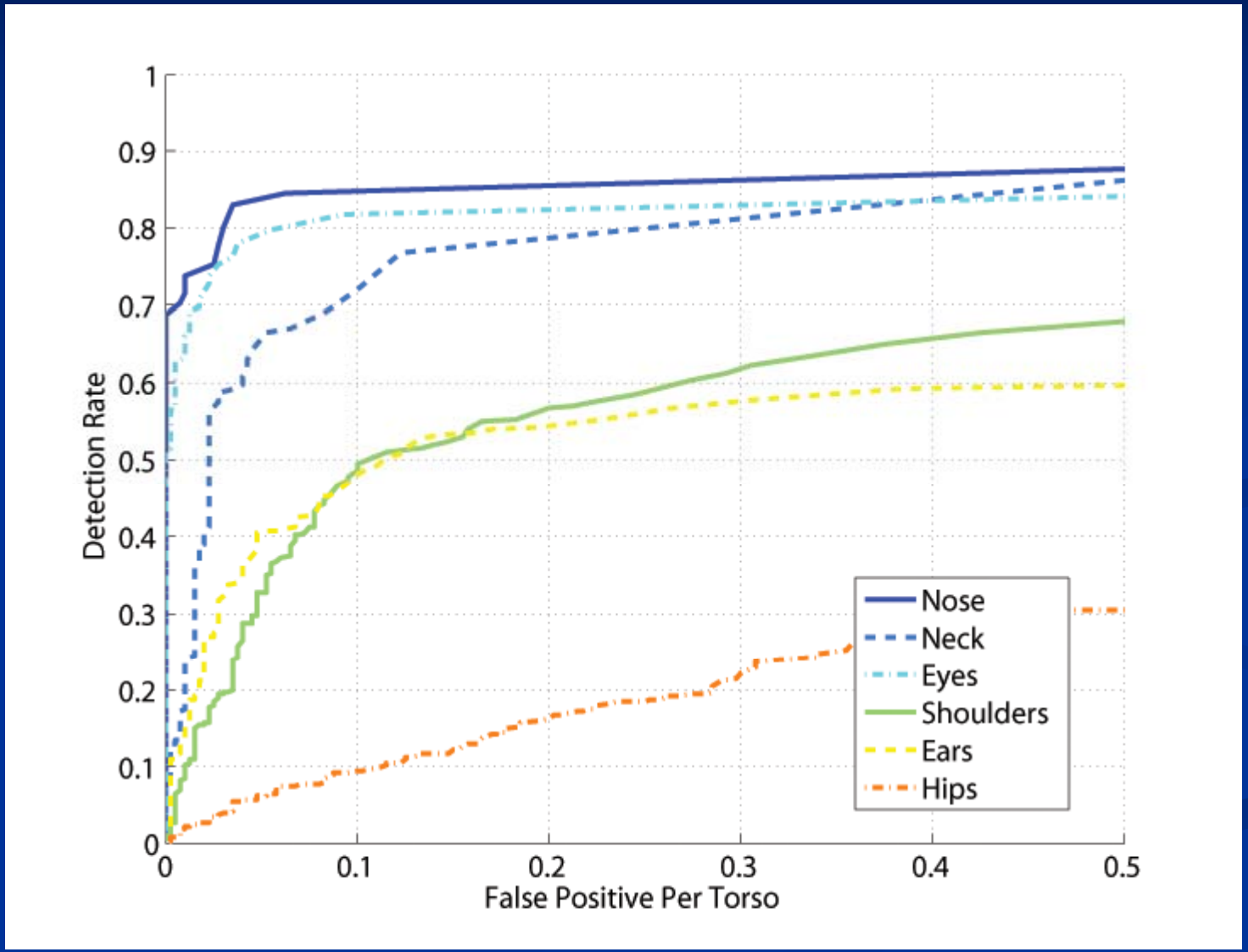


- Detect poselets
- Hough-vote for each keypoint location
- Cluster votes
- Score each cluster:

$$S(x) = \sum_i w_i a_i(x)$$

$w_i \propto$ How well poselet i predicts the right eye

Keypoint Detection ROC



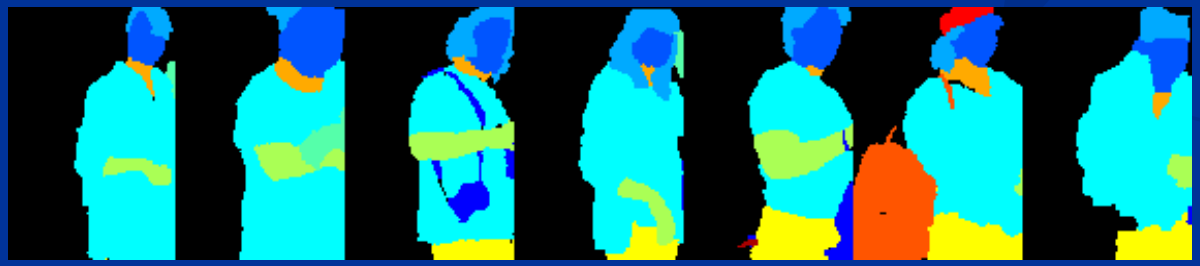
Agenda

- What is a Poselet
- How do we construct poselets?
- How do we use poselets for:
  - Detection
  - 3D Pose localization
  - Segmentation
- Results on VOC
- Conclusion

Poselets for Segmentation

- Examples of a poselet:

- H3D supplies region labels:



Expected "Person" label

- Except we didn't have time to train M^2HT

VOC Results (Person)

Detection	Poselets (Comp 4)	Best in Comp 3+4*
VOC 2009	**43.2%**	41.5%
VOC 2008	**48.7%**	47.6%

*** UoCTTI_LSVM-MDPM (P. Felzenszwalb, R. Girshick, D. McAllester)**

Segmentation	Poselets (Comp 6)	Best in Comp 5+6*
VOC 2009	36.3%	**38.9%**
VOC 2008	37.8%	**41.3%**

*** UCI_LAYEREDSHAPE (C. Fowlkes, S. Hallman, D. Ramanan, Yi Yang)**

Training Data Comparison

- We used:
  - VOC09 trainval: For M^2HT and bounds regression
  - H3D: For poselet selection and training
- Number of positive training images:
  - VOC09 trainval: 2779
  - H3D: 260
- Number of annotated people:
  - VOC09 trainval: 5815
  - H3D: 750

The "supervision" line

This work

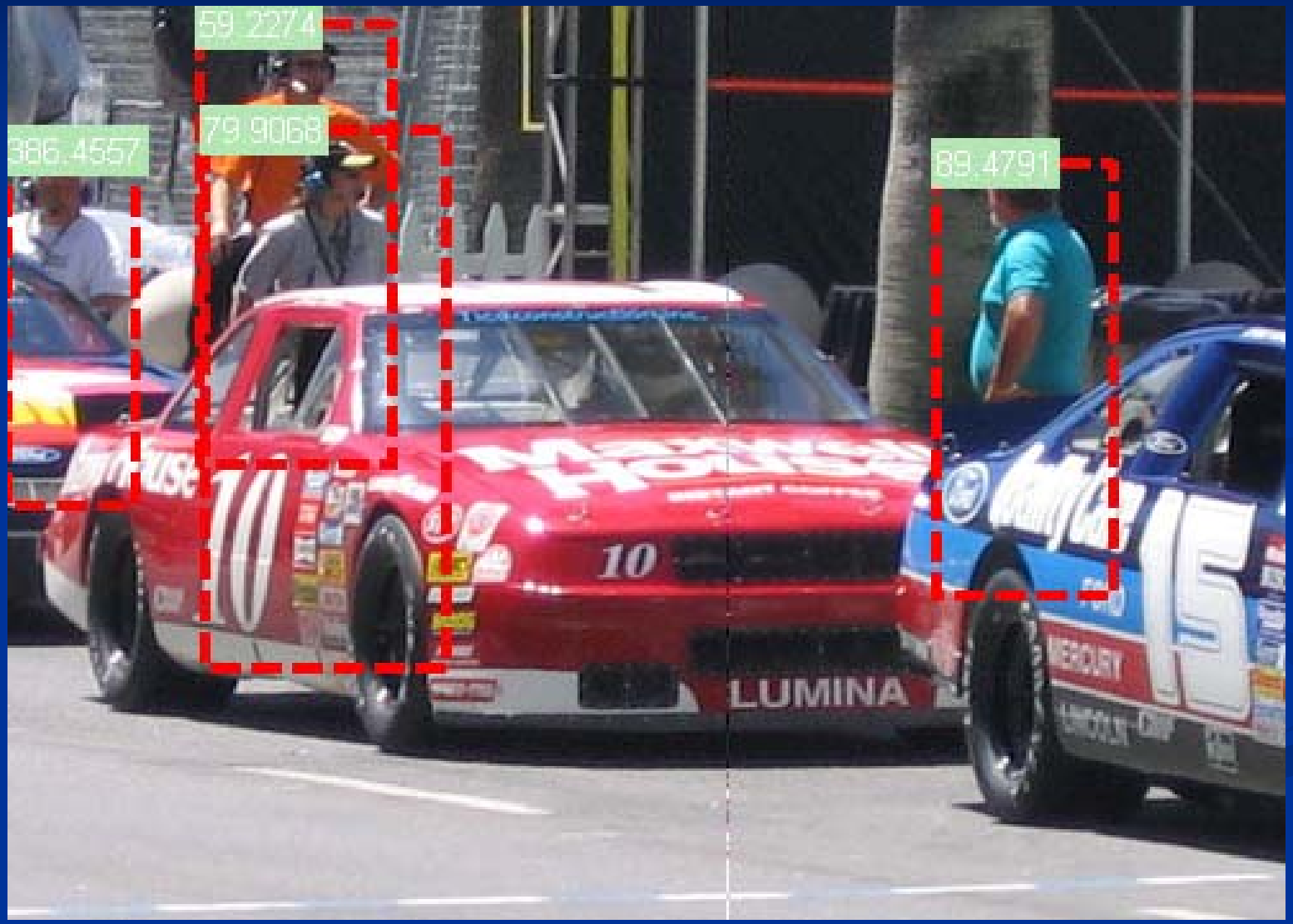
Unsupervised

Weakly Supervised

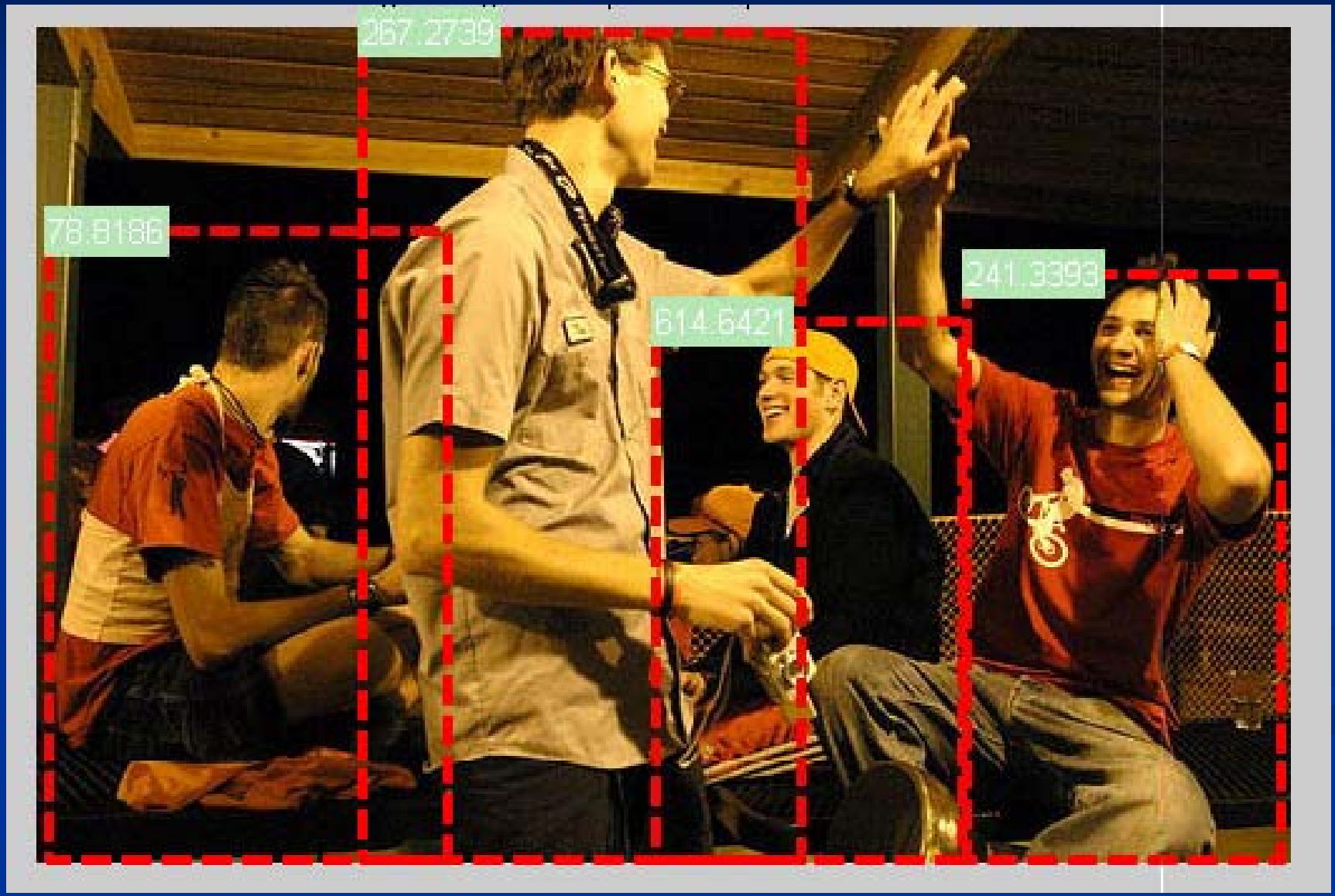
Supervised

Strongly Supervised

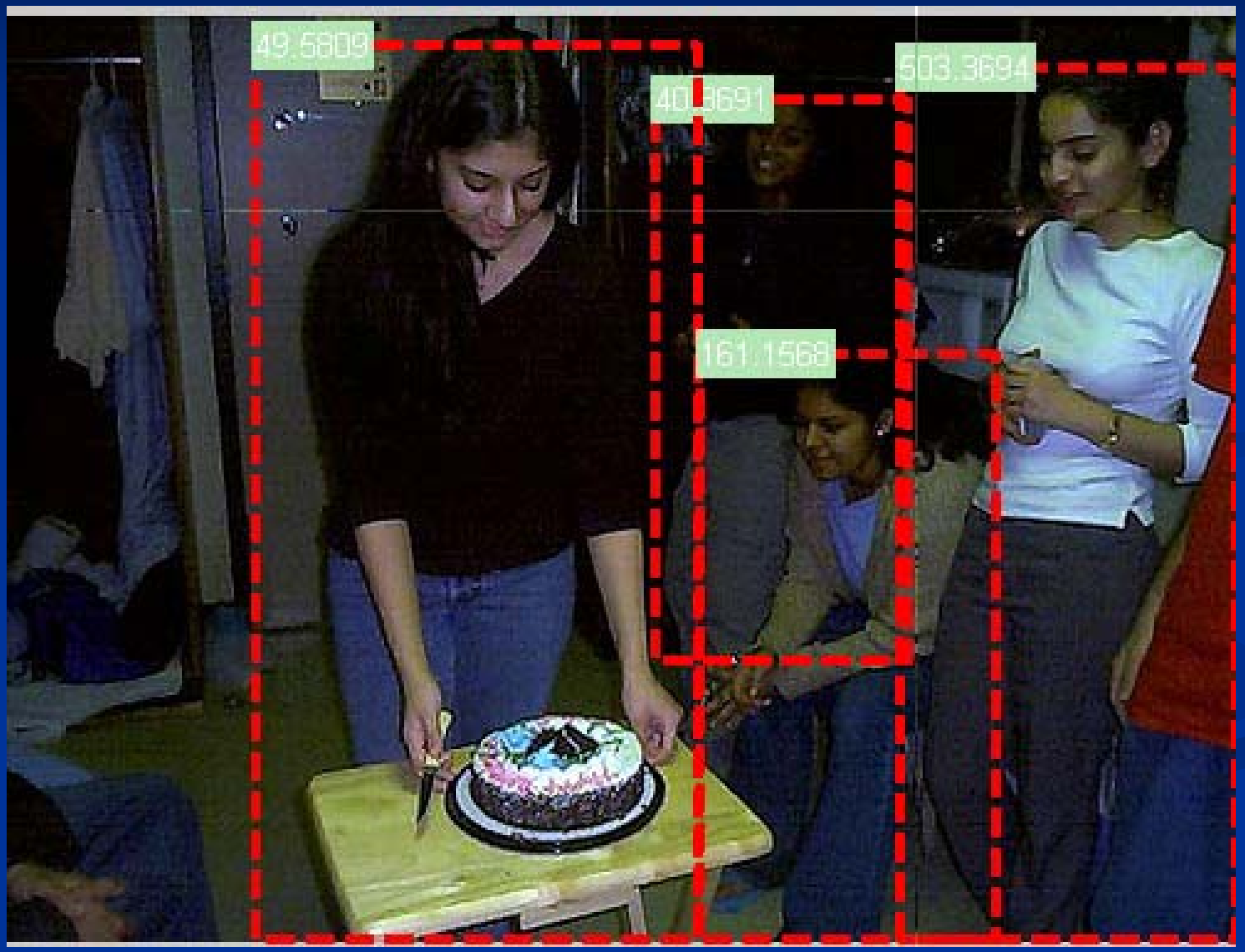
Detection results



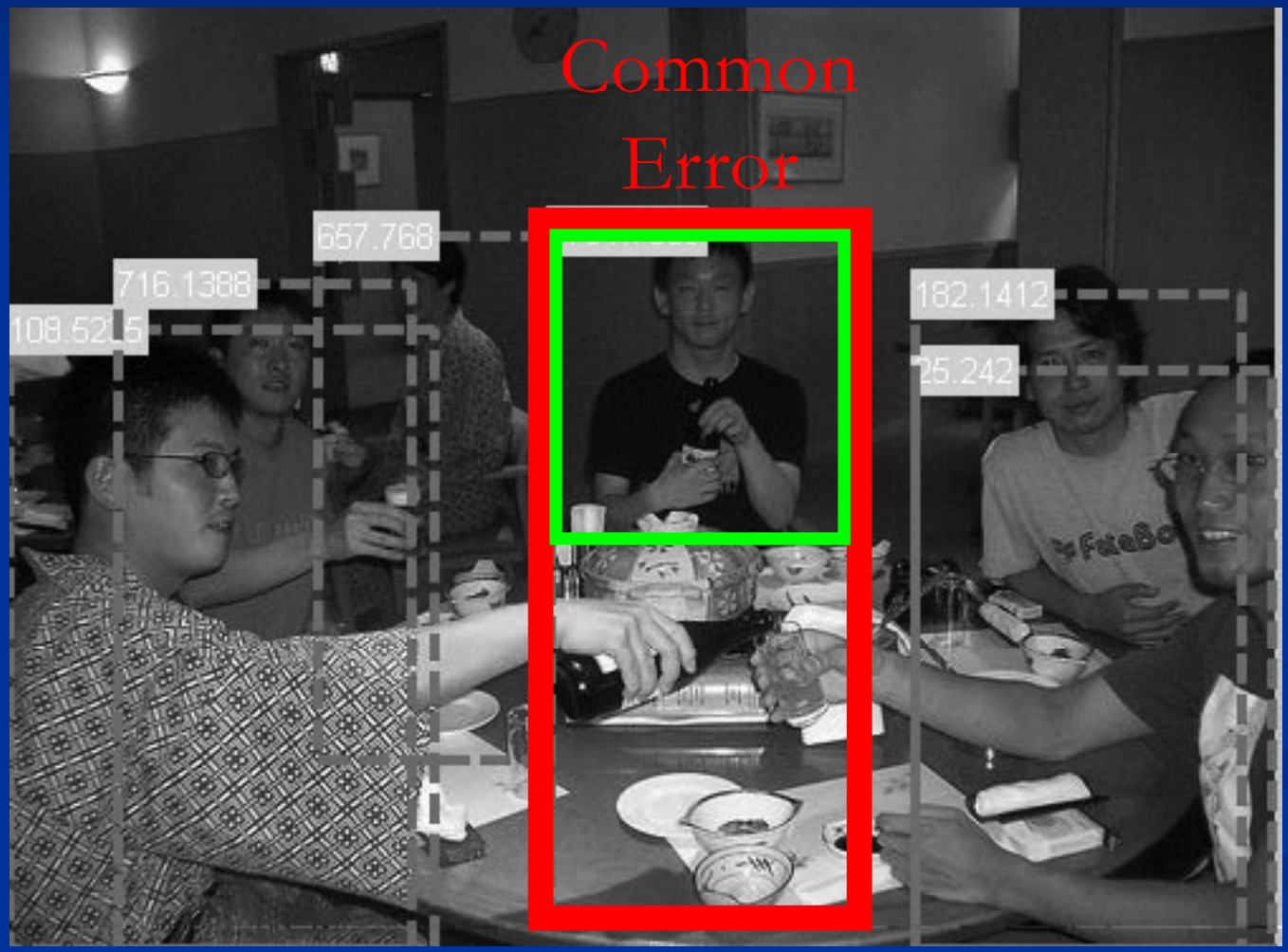
Detection results



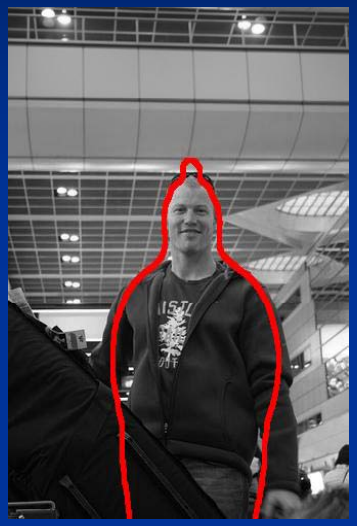
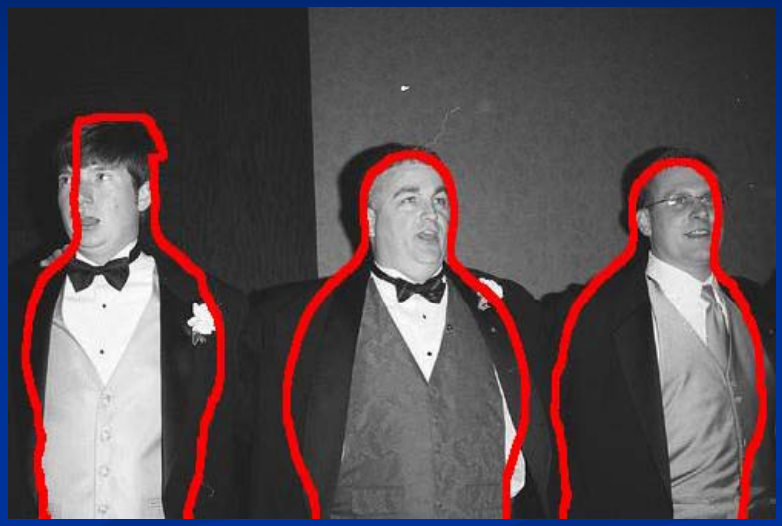
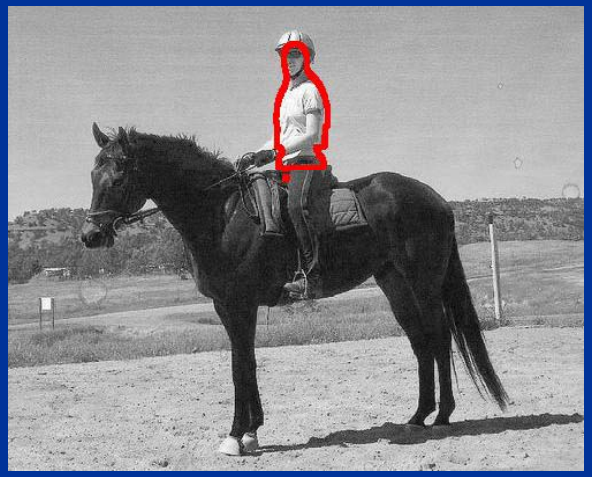
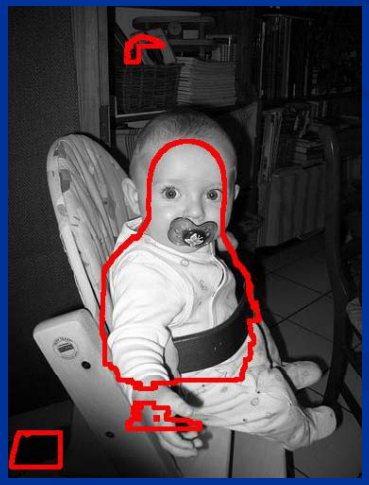
Detection results



Detection results



Segmentation Examples on VOC09


Segmentation Examples on VOC09

What about horses?

Horse poselets (preliminary results)

Conclusion

- Poselets: Novel parts that provide a bridge between appearance and configuration space.
- Poselets are effective for:
  - Detection
  - Localization
  - Segmentation
- Naturally extend to other visual categories

Poselets Web Site

http://eecs.berkeley.edu/~lbourdev/poselets

- The paper
- H3D data set + Matlab tools
- Java3D annotation tool + video tutorial
- Matlab code to detect people using poselets
- Our latest trained poselets

All available free with a non-commercial license

BROWN

Body Shape from Video

Alexandru Balan Michael Black
Brown University

BROWN

What's constant?

Constant:

- camera pose and focal length
- identity
- height, weight
- limb lengths
- global scene illumination
- albedo

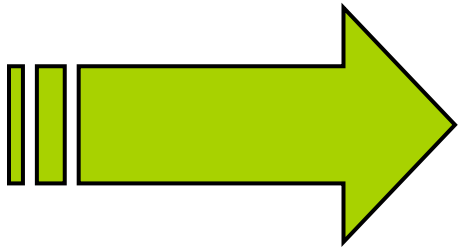
Changing

- pose (joint angles and spine)
- visibility
- soft tissues
- drape of clothing
- cast shadows and shading

BROWN

What's constant?

Approach: Model what's constant using a model of 3D body shape.



Problem: Pose changes shape. Need a pose-invariant shape model.

Why a graphics model?

Goals

- Provide strong constraints for interpreting video.
- Combine information
  - Across views
  - Across non-rigid pose changes
- Support inference of gender, height, age, etc.
- Explain changes due to illumination.

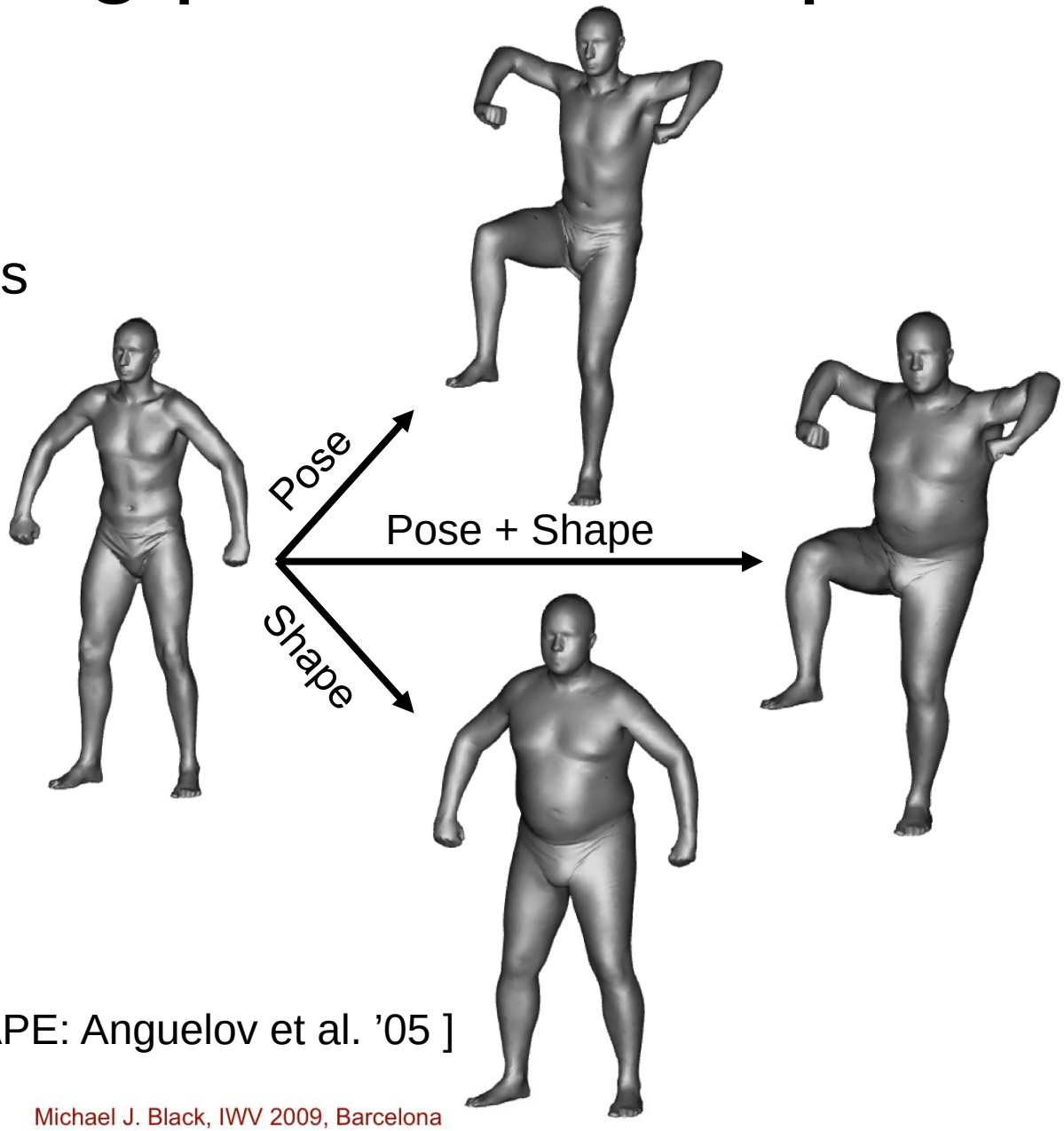
Problem: must factor changes due to shape and pose

- Learn statistics of shape variation across people and poses
- Model what you know, learn the statistics of the rest.

Factoring pose and shape

Low dimensional parameterization learned from examples

We use an "intrinsic" shape representation *invariant* to pose



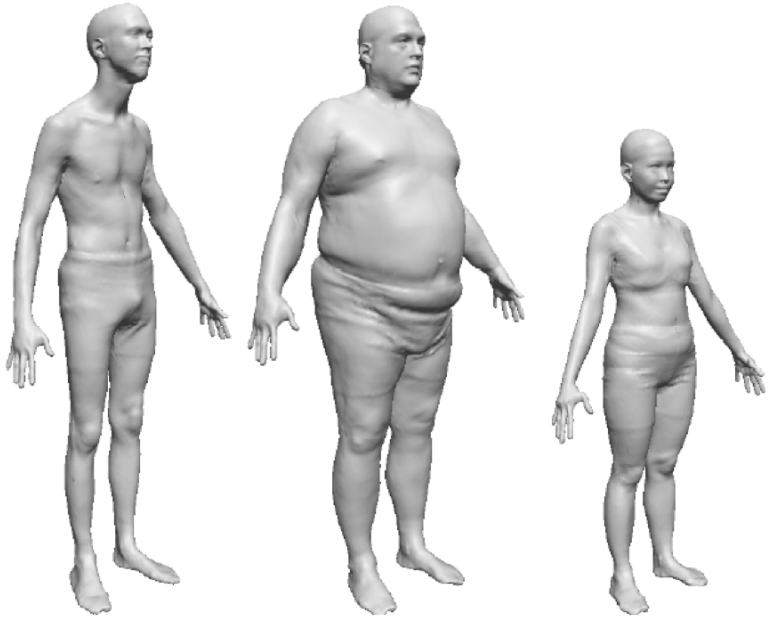
[SCAPE: Anguelov et al. '05]

SCAPE

Shape Completion and Animation of PEople

[Anguelov et al. Siggraph ’05]

[Cyberware]

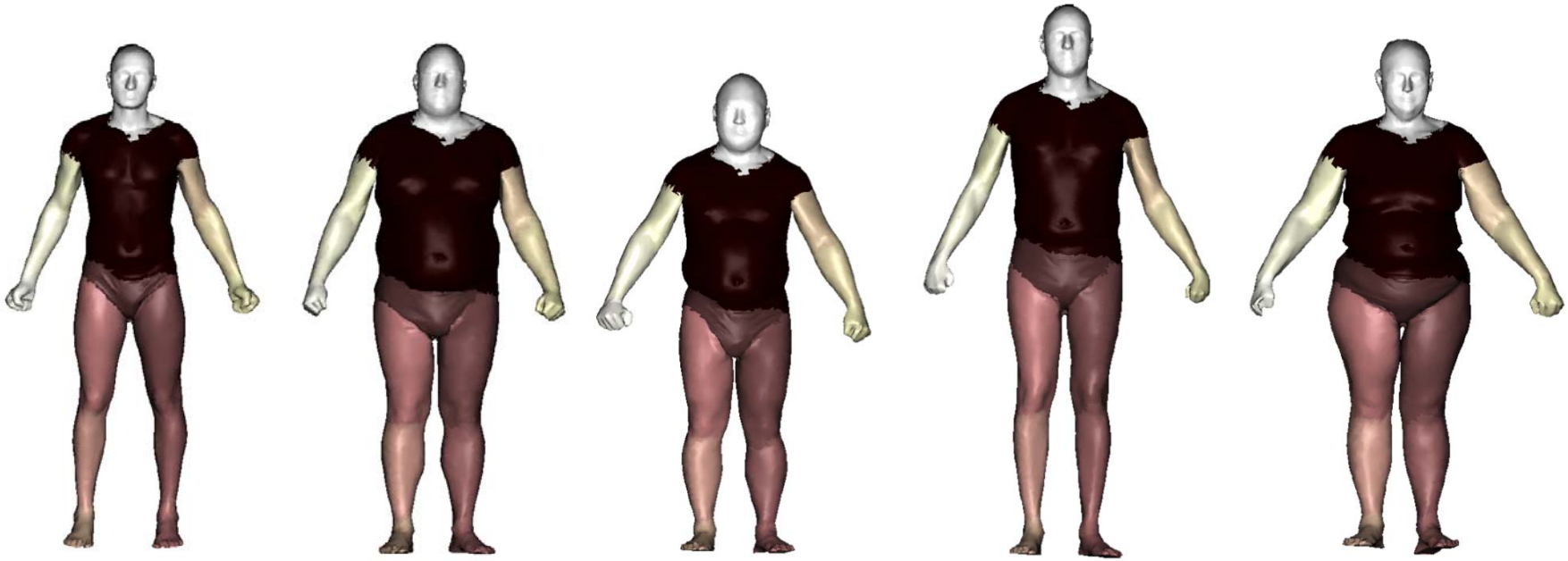


[Allen et al. ’03]

CAESAR dataset (SAE Int.): 3D mesh models of over 2000 North American adults

Shape space

- Align a “template mesh” to each scan using an iterative closest point method.
- 25,000 triangles and 12,500 vertices.

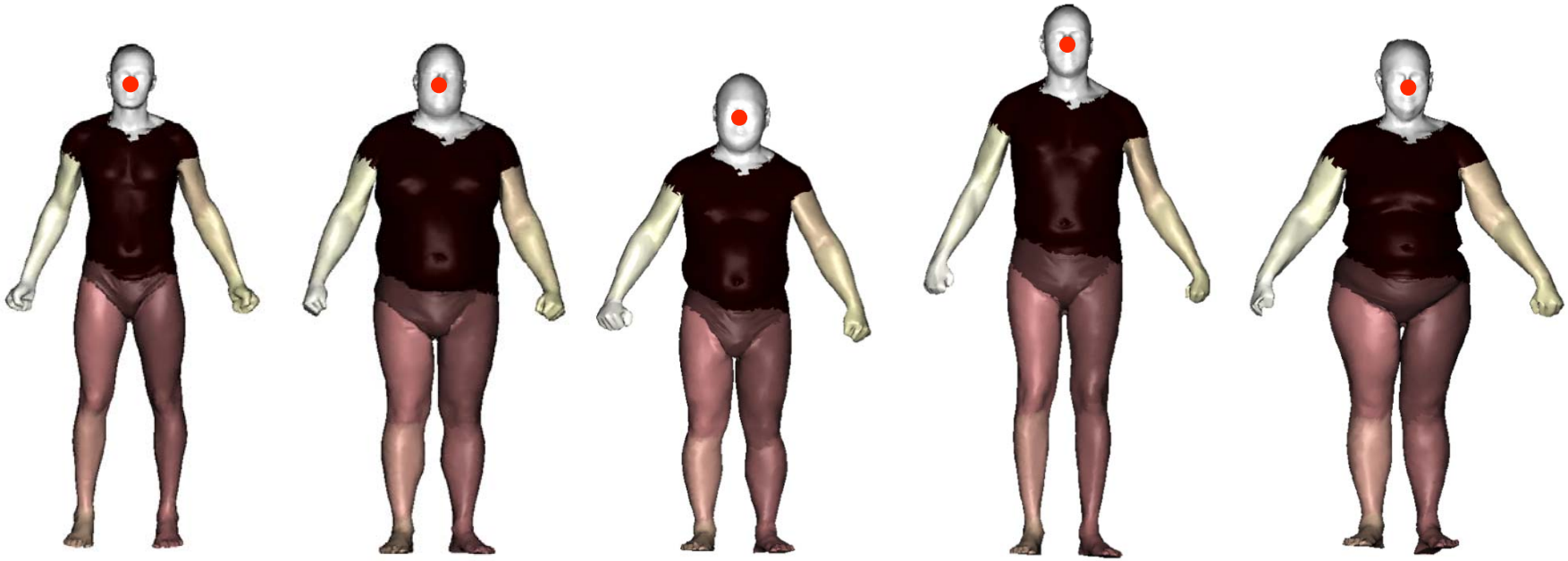


Example scans of different people.

[Anguelov et al. ’05]

Shape space

- All vertices are in correspondence.
- All bodies in the canonical pose.
- Vertices assigned to body “parts”.

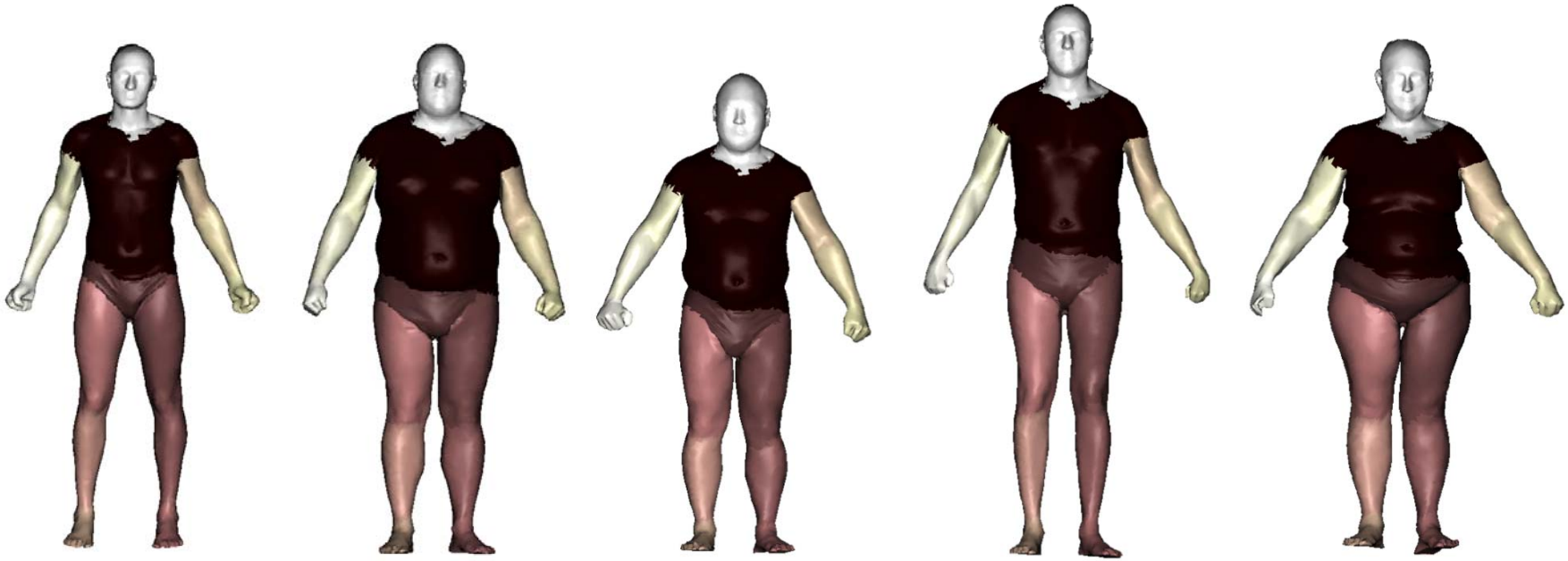


Example scans of different people.

[Anguelov et al. ’05]

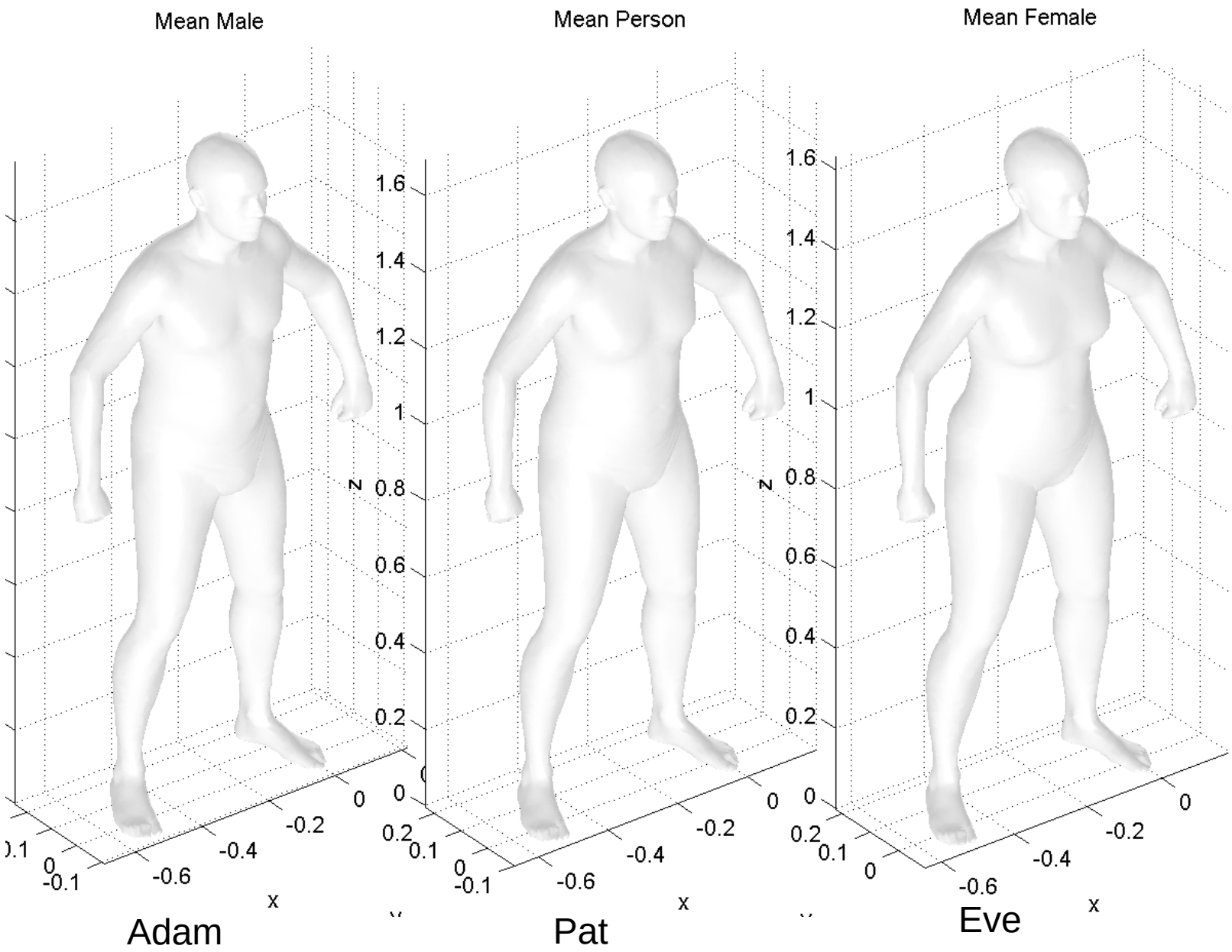
Shape space

- Learn low dimensional shape deformation model using incremental PCA (Brand, ECCV'02).
  - applied to deformations (i.e. rotations) of the triangles.



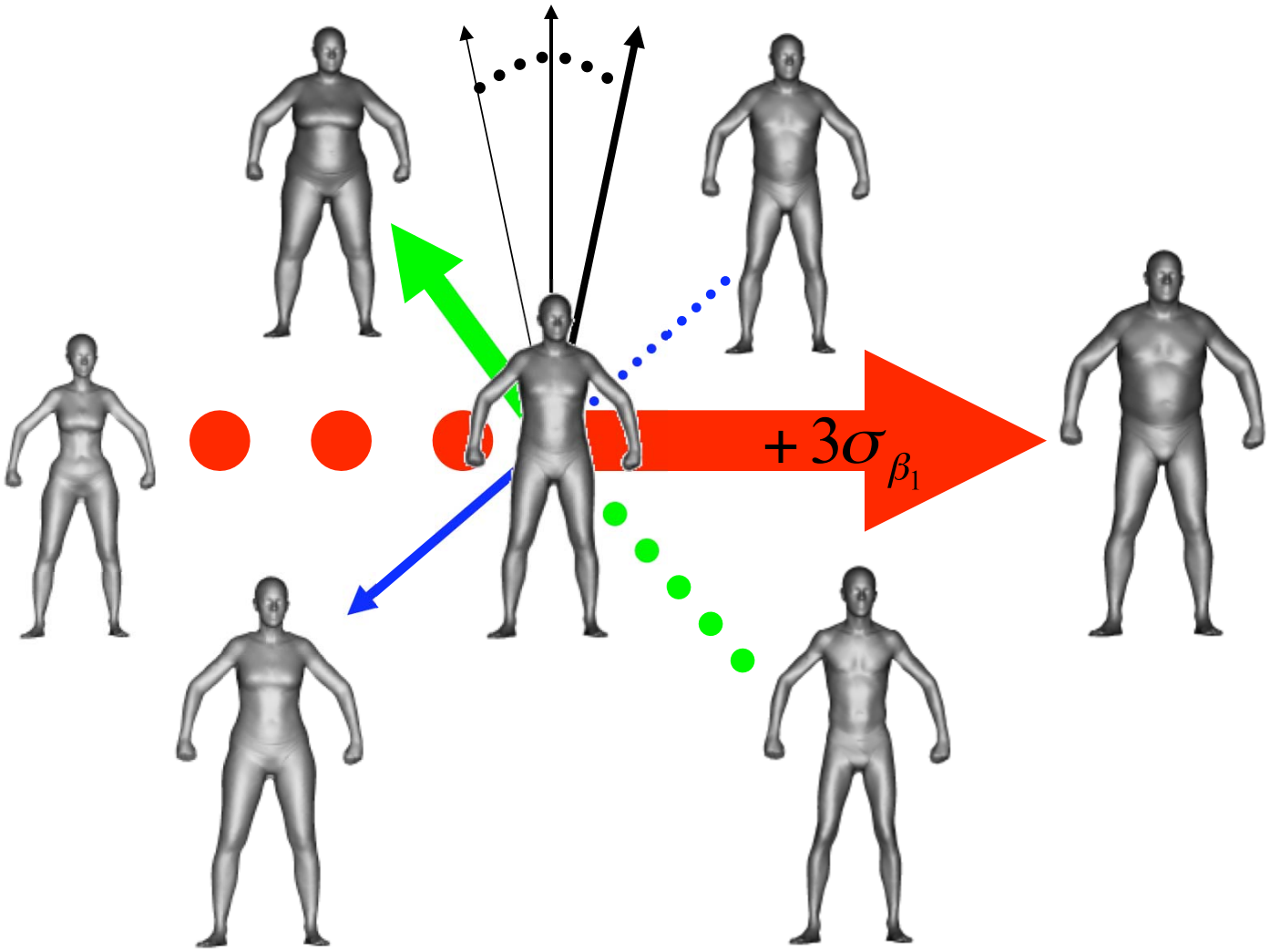
Example scans of different people.

[Anguelov et al. '05]


Mean Male
Mean Person
Mean Female
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
z
0
-0.2
-0.4
-0.6
x
0.2
0.1
0
-0.1
y
Adam
Pat
Eve

BROWN

Eigen-People



What about pose changes?

For each part, apply a rigid rotation to each triangle.

R_{part}

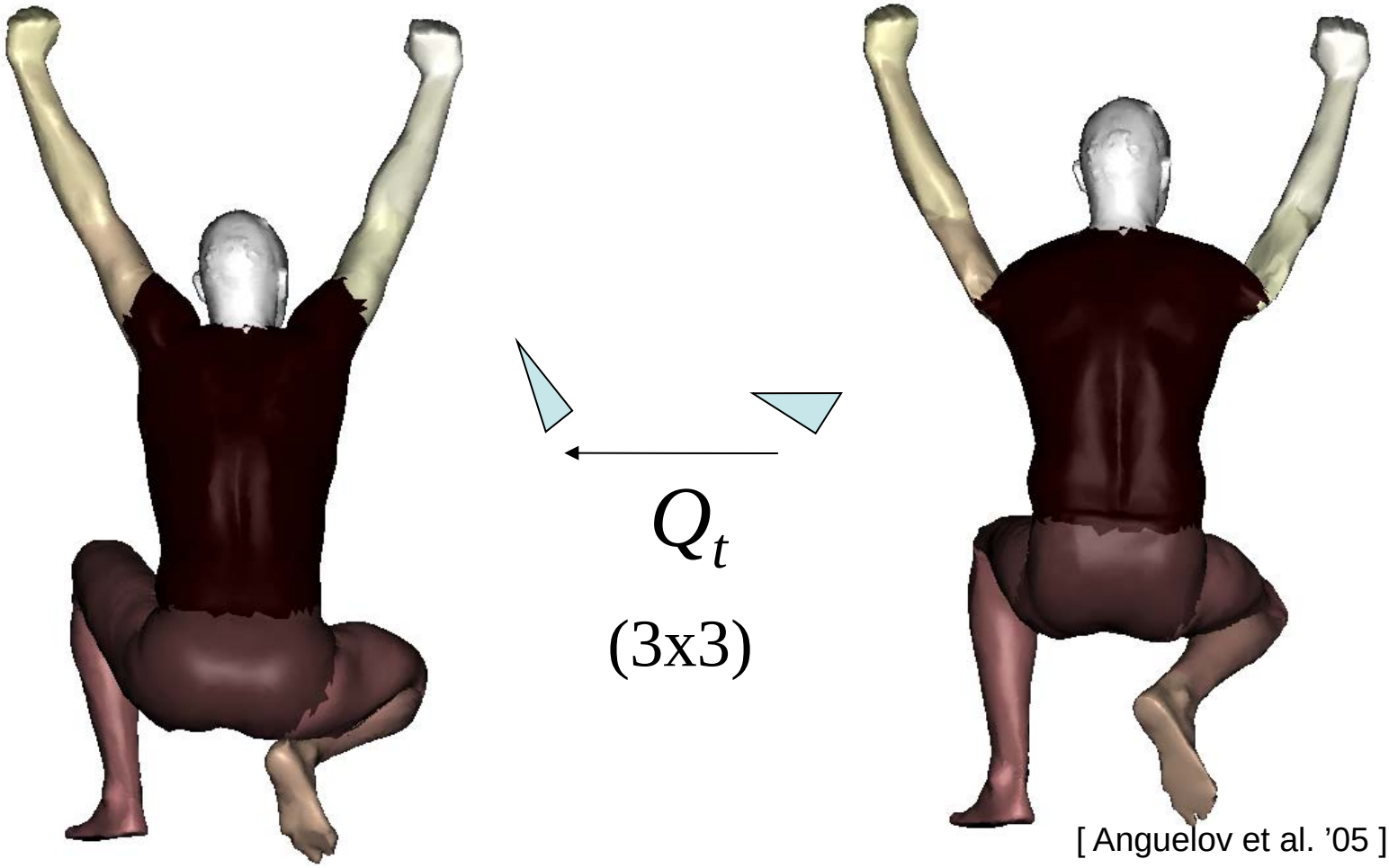
(3x3)
x 15 parts

[Anguelov et al. ’05]

Preserve desired orientation and scale of edges in a least-squares sense

What about pose changes?

Triangles are all in correspondence.



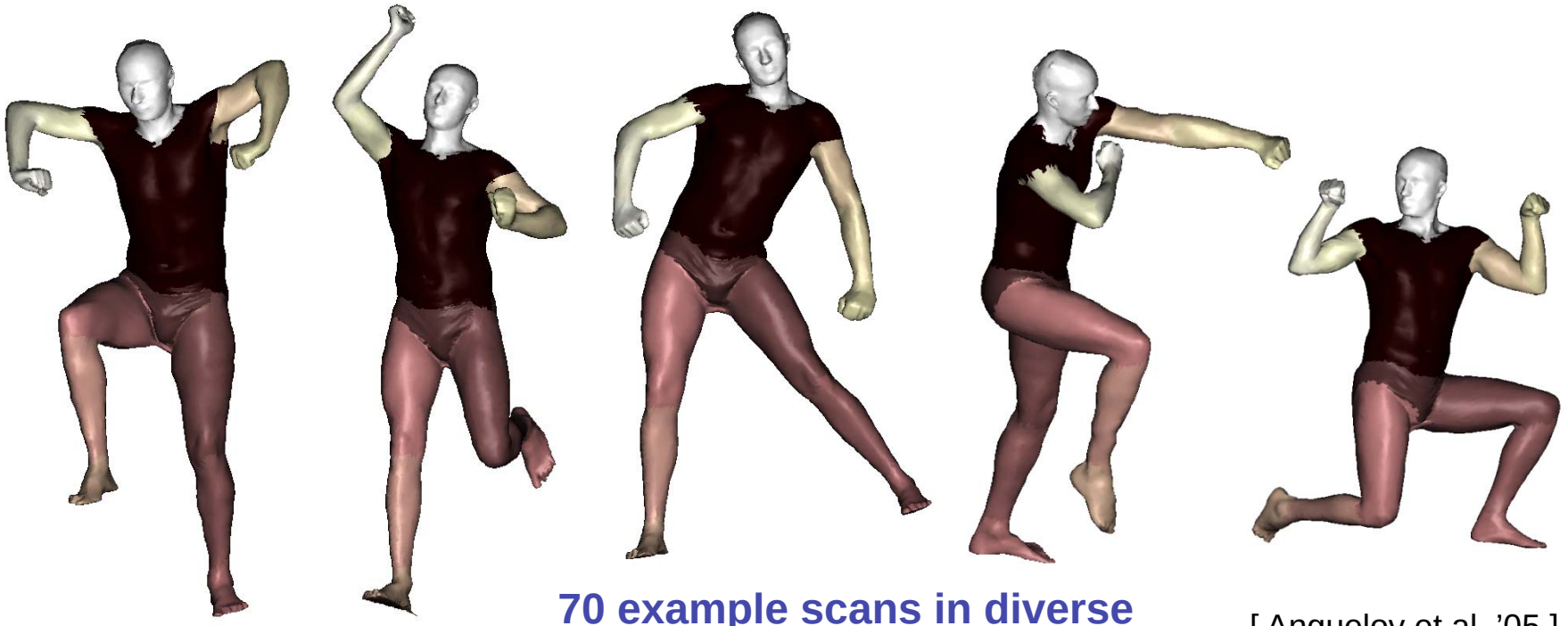
[Anguelov et al. ’05]

Deformation of each triangle from predicted model to training example.

Pose deformation space

Model articulated and non-rigid deformations

– Non-rigid deformations learned as a function of relative part orientations.

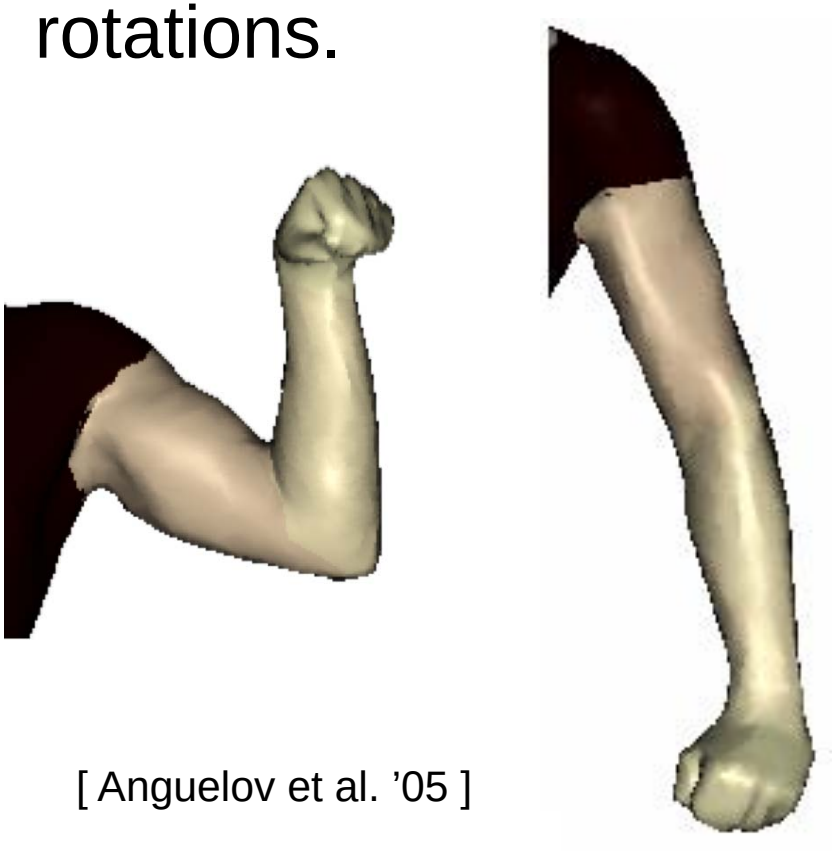


70 example scans in diverse poses

[Anguelov et al. ’05]

Pose deformation space

- Model non-rigid deformations.
- Linear prediction from relative part rotations.

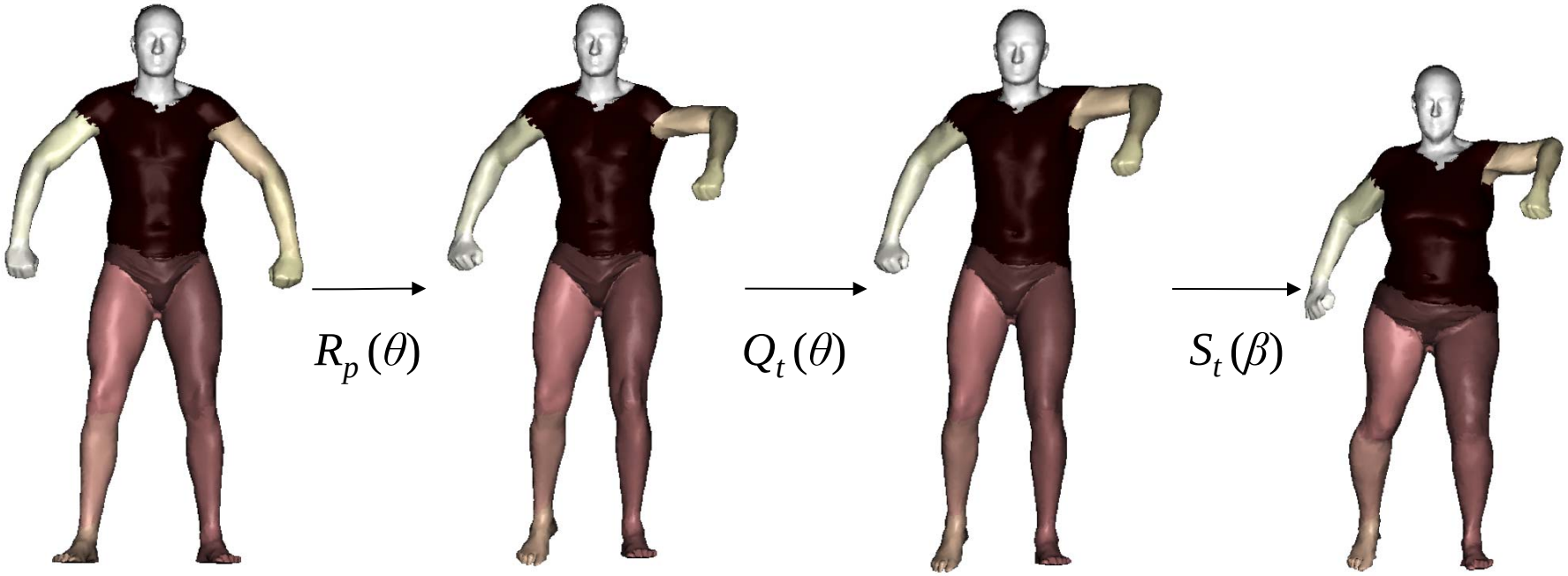


[Anguelov et al. ’05]

$$Q_t = A_t \cdot \begin{bmatrix} \theta_{p_t^1} \\ \theta_{p_t^2} \\ 1 \end{bmatrix}$$

Learn a matrix A_t for each triangle t in the mesh via linear regression.

SCAPE deformations



Articulated Rigid Deformation

Non-rigid Deformation

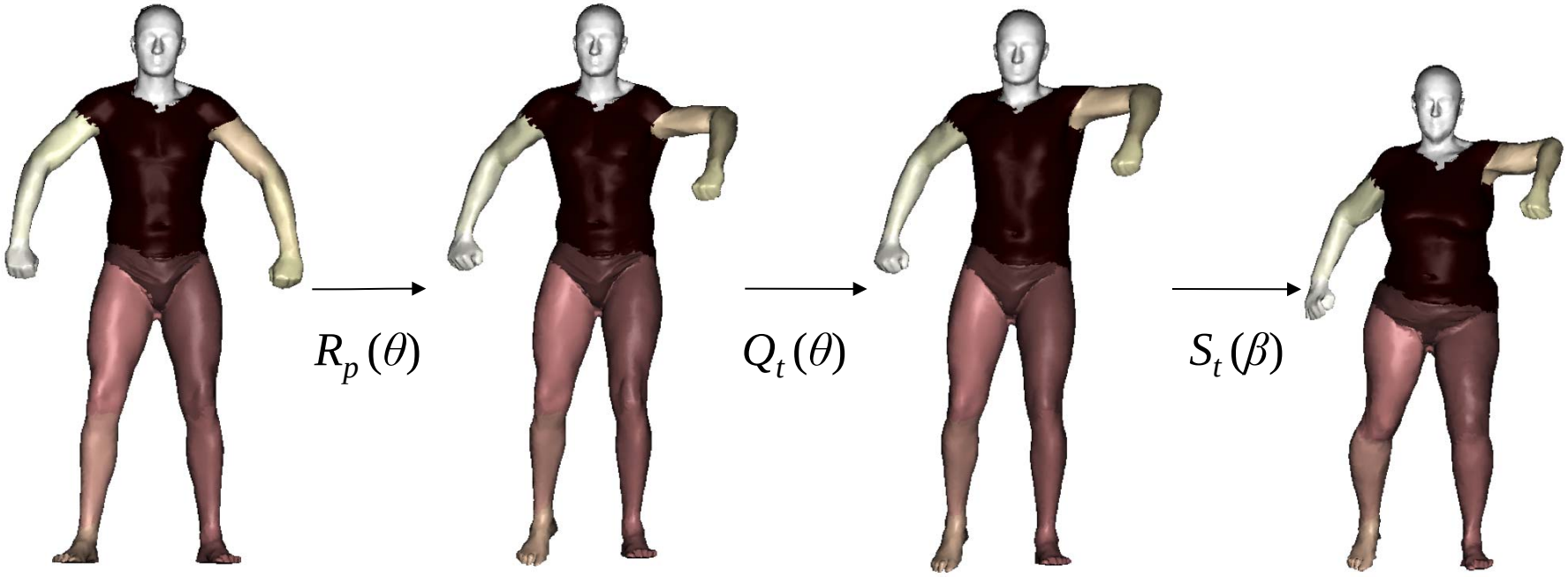
Body Shape Deformation

θ – part rotations

β – shape parameters

[Anguelov et al. '05]

SCAPE deformations

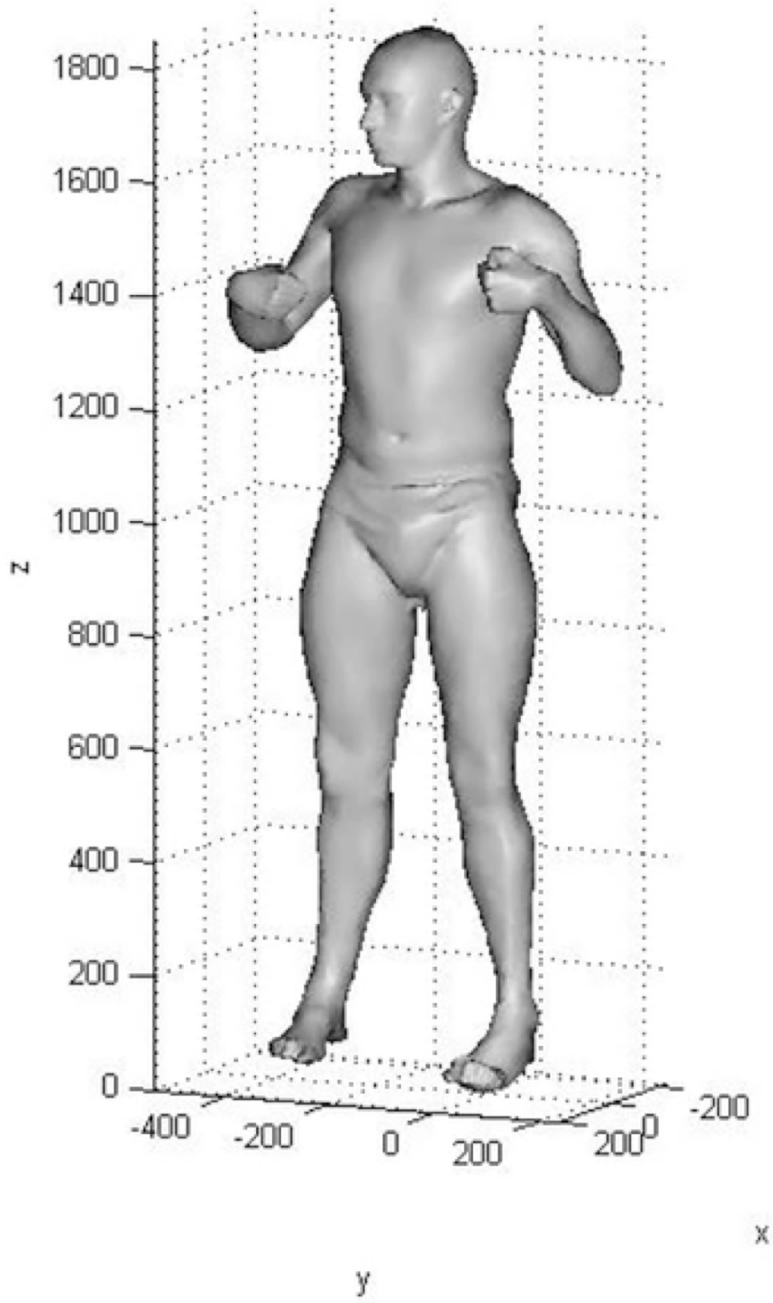


θ – part rotations (37D) τ – global position (3D) β – shape parameters (6-20D)

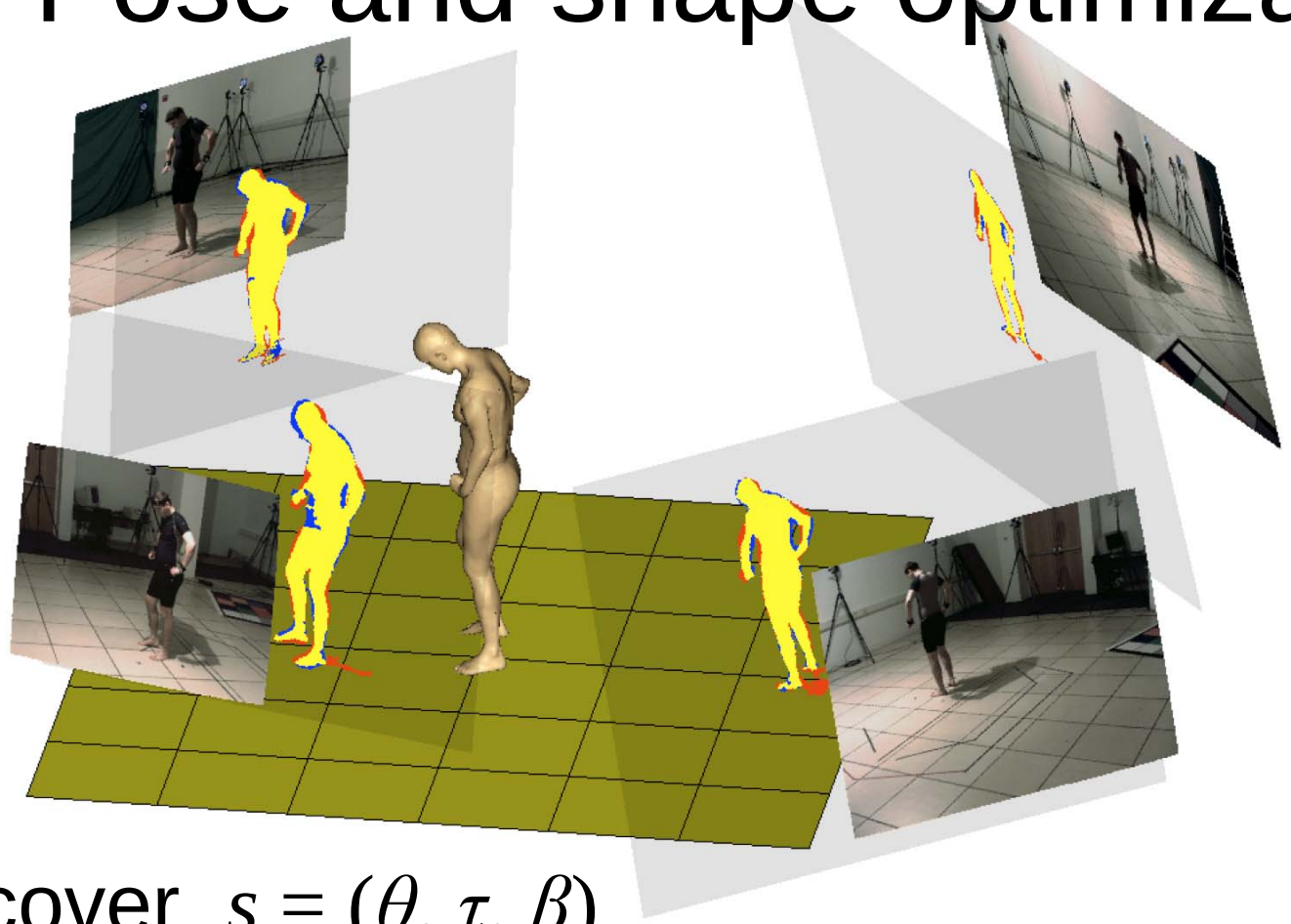
Parameters (state): $s = (\theta, \tau, \beta)$

Shape parameters can be gender specific.

[Anguelov et al. '05]


1800
1600
1400
1200
1000
800
600
400
200
0
z
-400
-200
0
200
y
200
0
-200
x

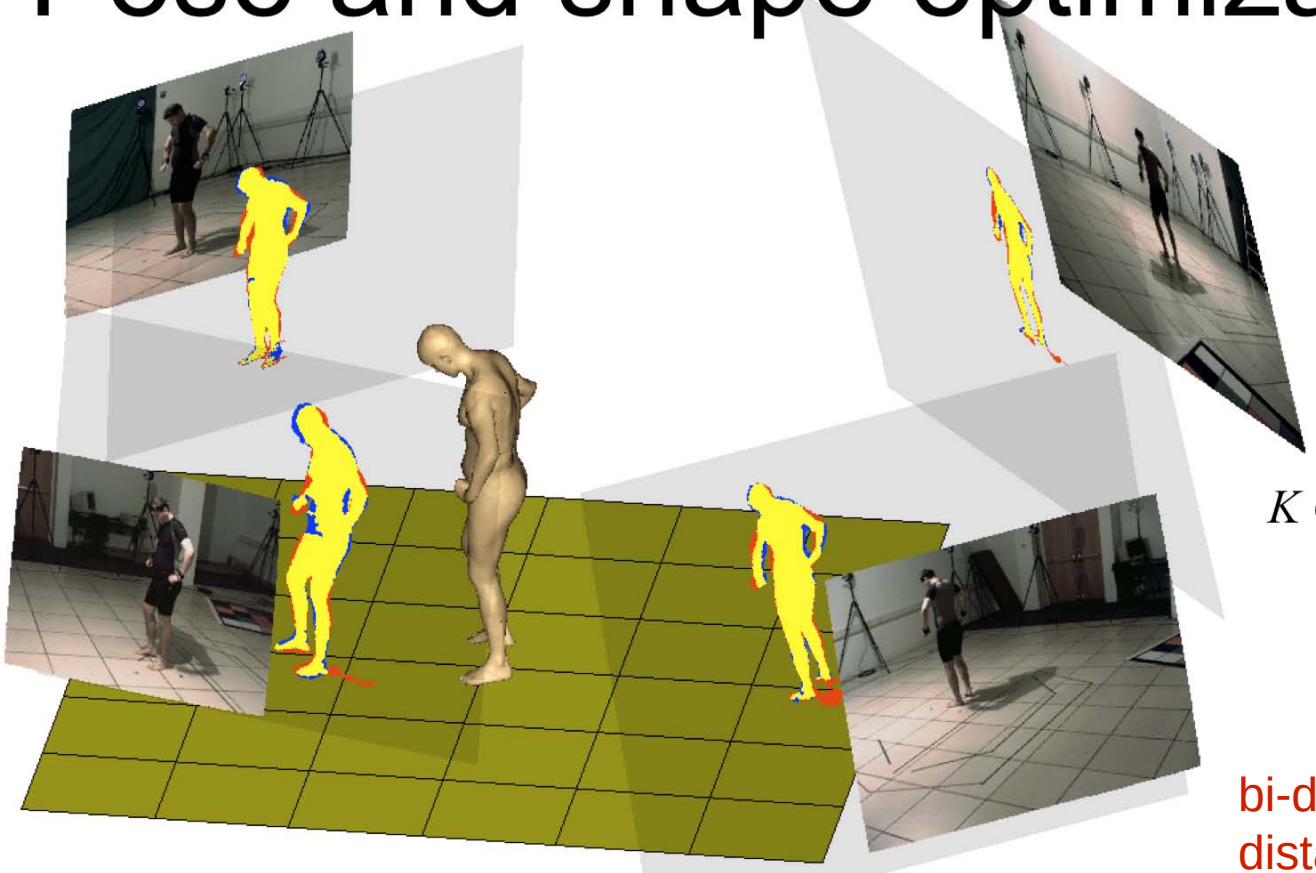
Pose and shape optimization



Recover $s = (\theta, \tau, \beta)$

(Initialization: Sigal *et al.* NIPS '07)

Pose and shape optimization



K cameras

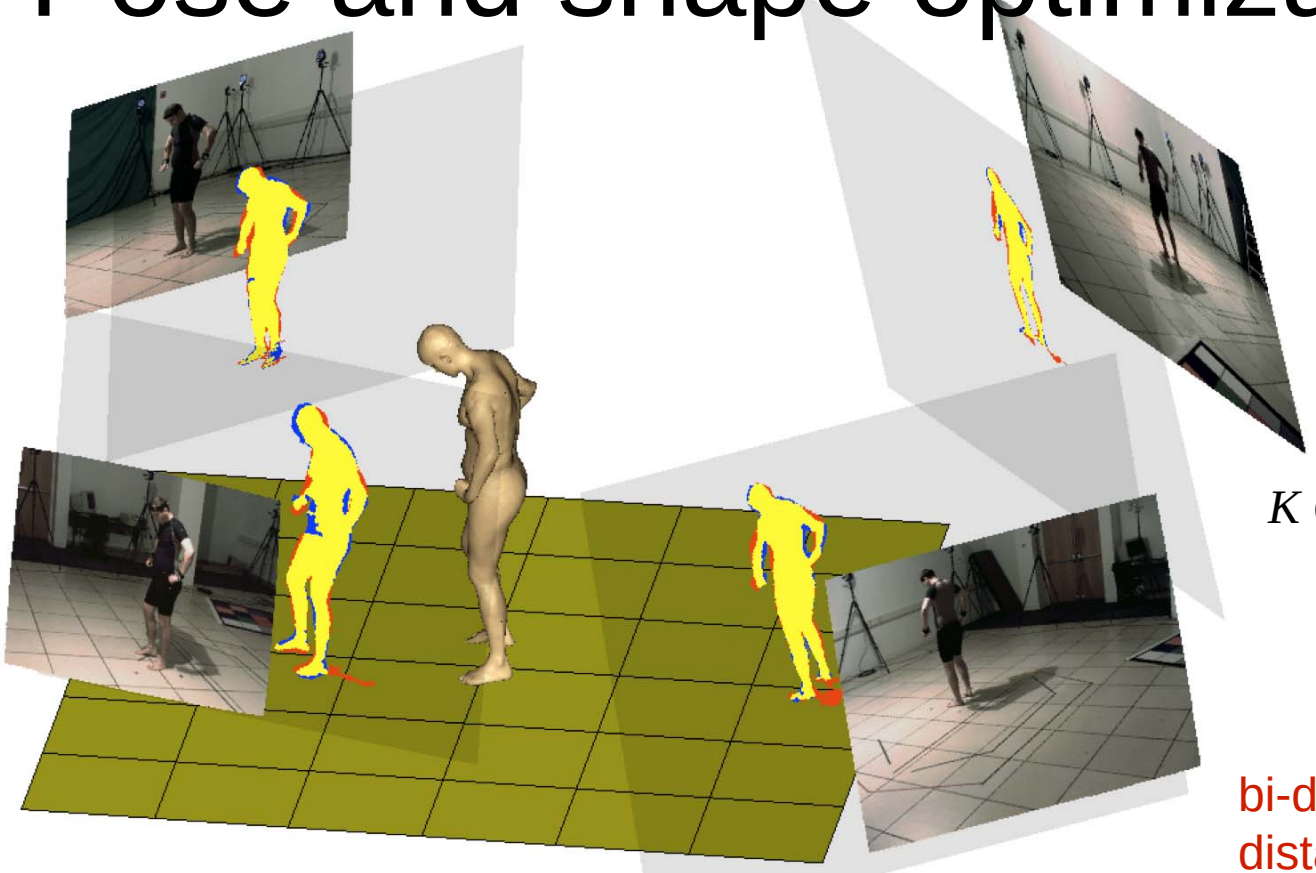
Recover $s = (\theta, \tau, \beta)$

bi-directional distance measure between silhouettes

Minimize $$E(s) = \sum_{i=1}^{K} D\left(\mathbf{F}_i^e(s), \mathbf{F}_i^o\right)$$

Penalize interpenetration.

Pose and shape optimization



K cameras

Recover $s = (\theta, \tau, \beta)$

bi-directional distance measure between silhouettes

Minimize $$E(s) = \sum_{i=1}^{K} \alpha\, d(\mathrm{F}_i^e(s), \mathrm{F}_i^o) + (1-\alpha)\, d(\mathrm{F}_i^o, \mathrm{F}_i^e(s))$$

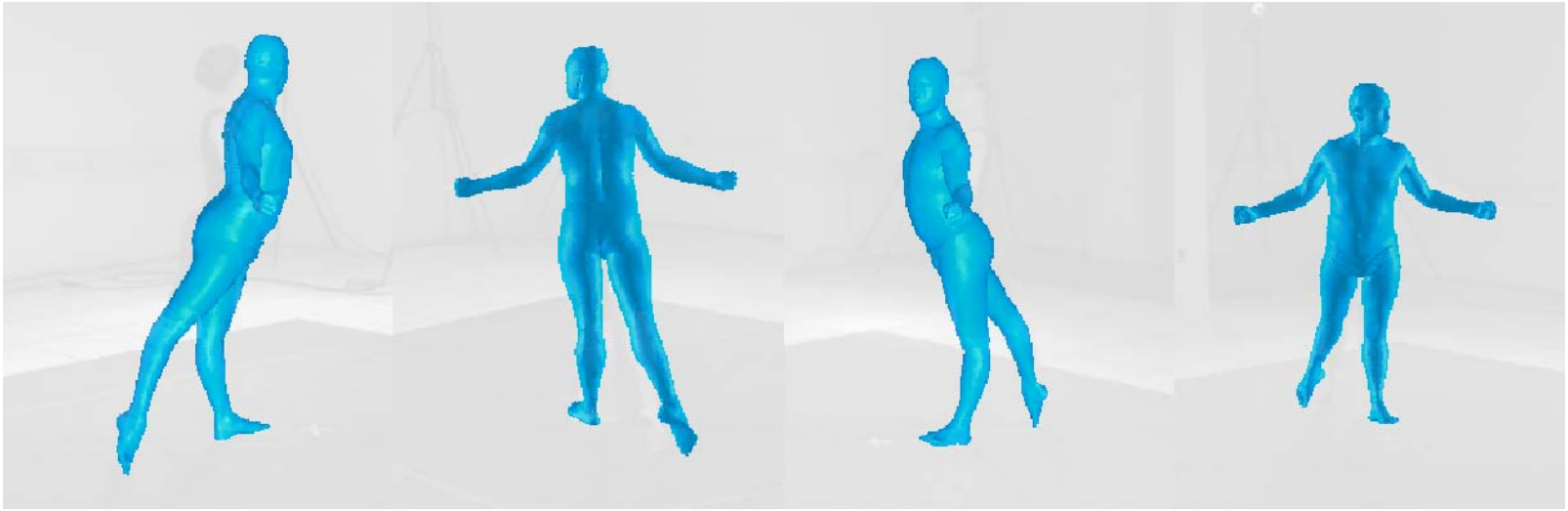
Penalize interpenetration.

BROWN

Optimization – Step by Step

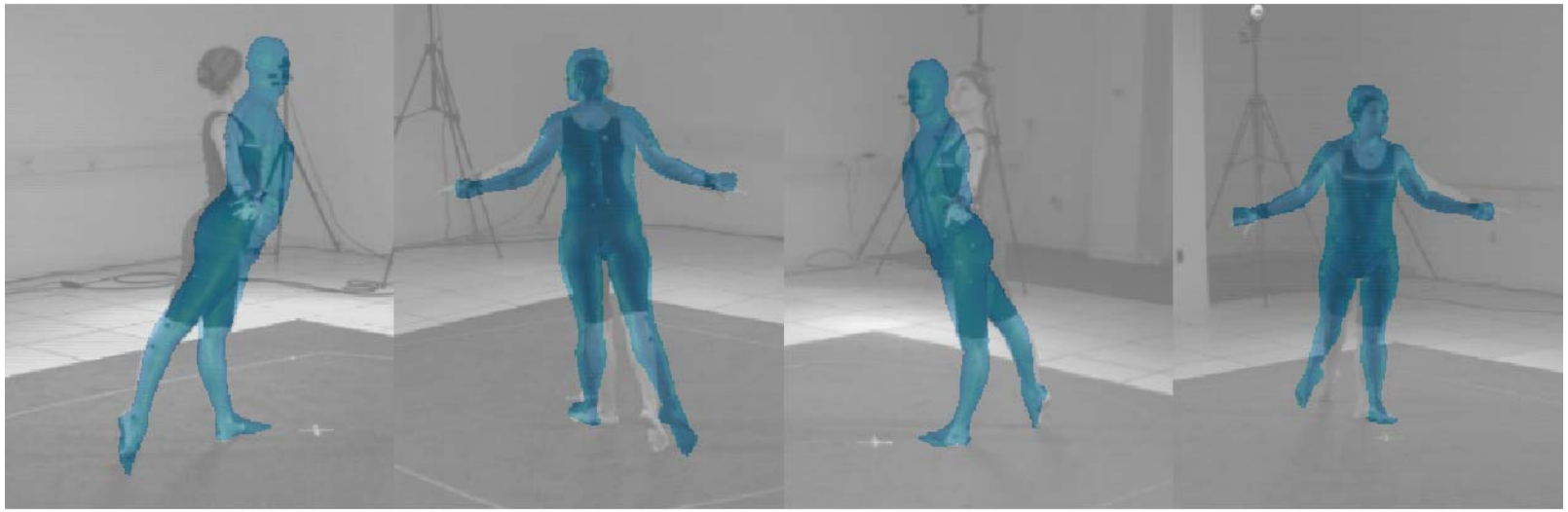
Input images and foreground silhouettes

Optimization – Step by Step

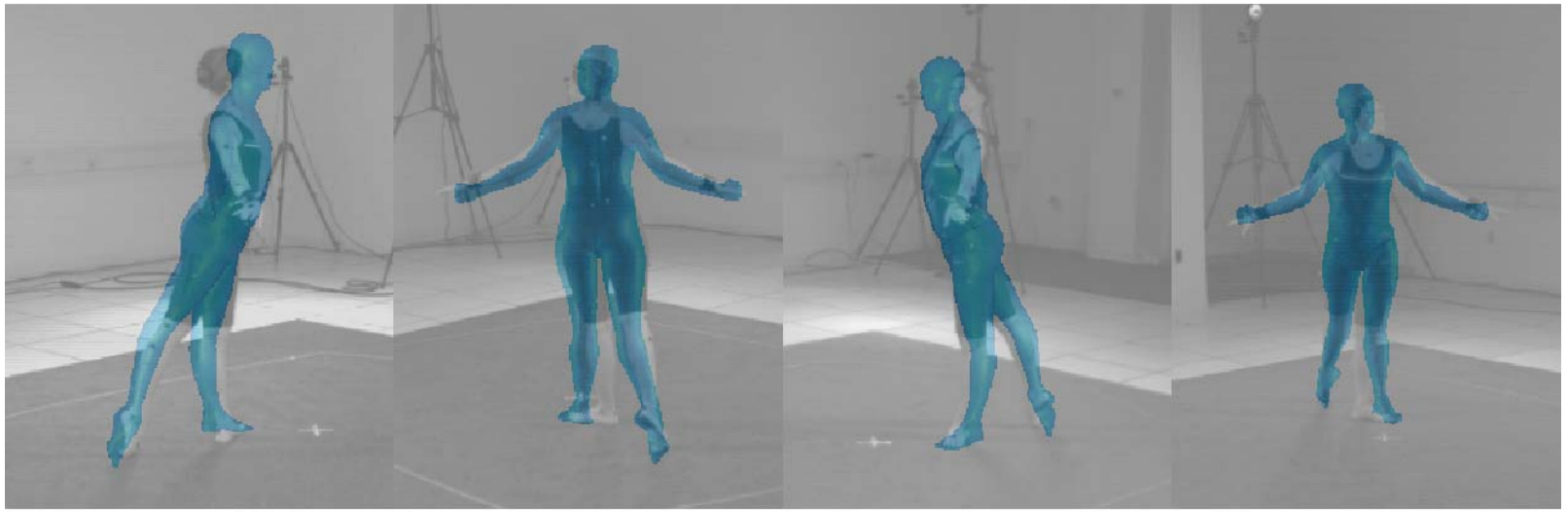


BROWN

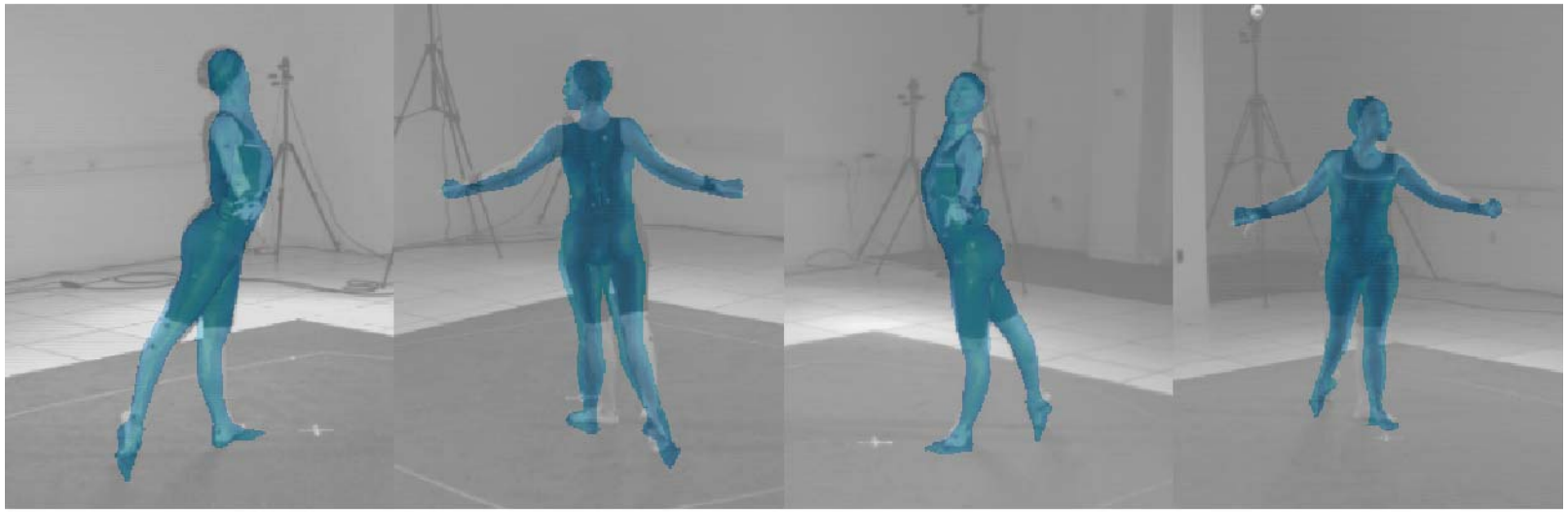
Optimization – Step by Step



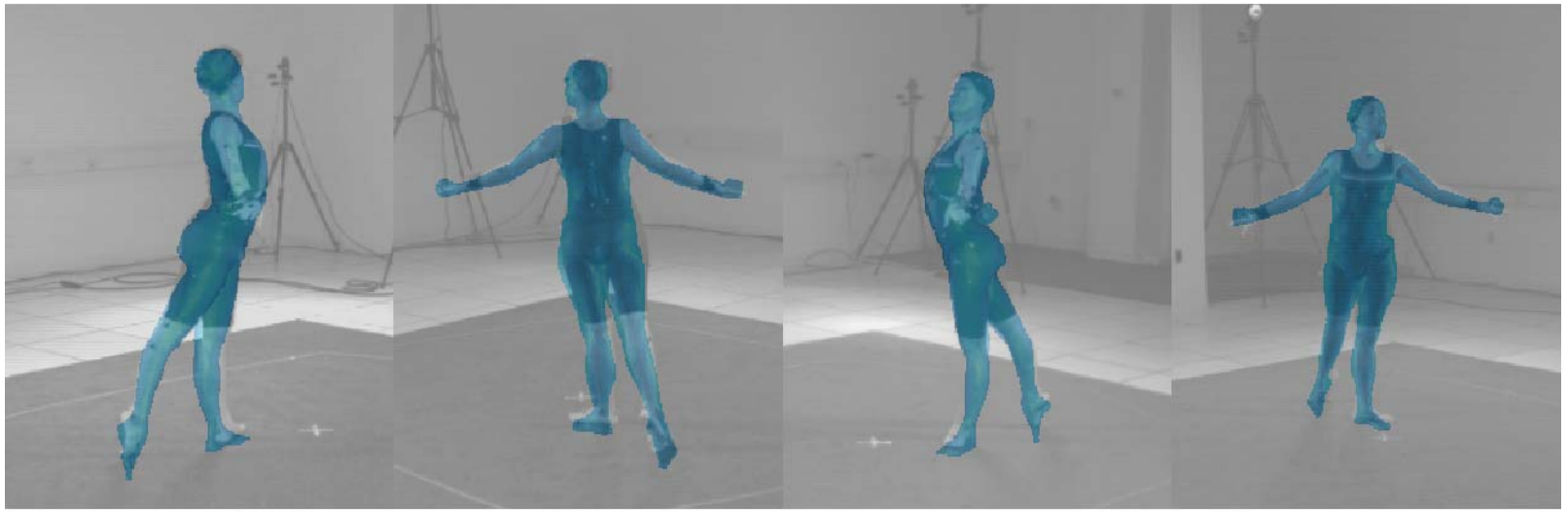
Optimization – Step by Step



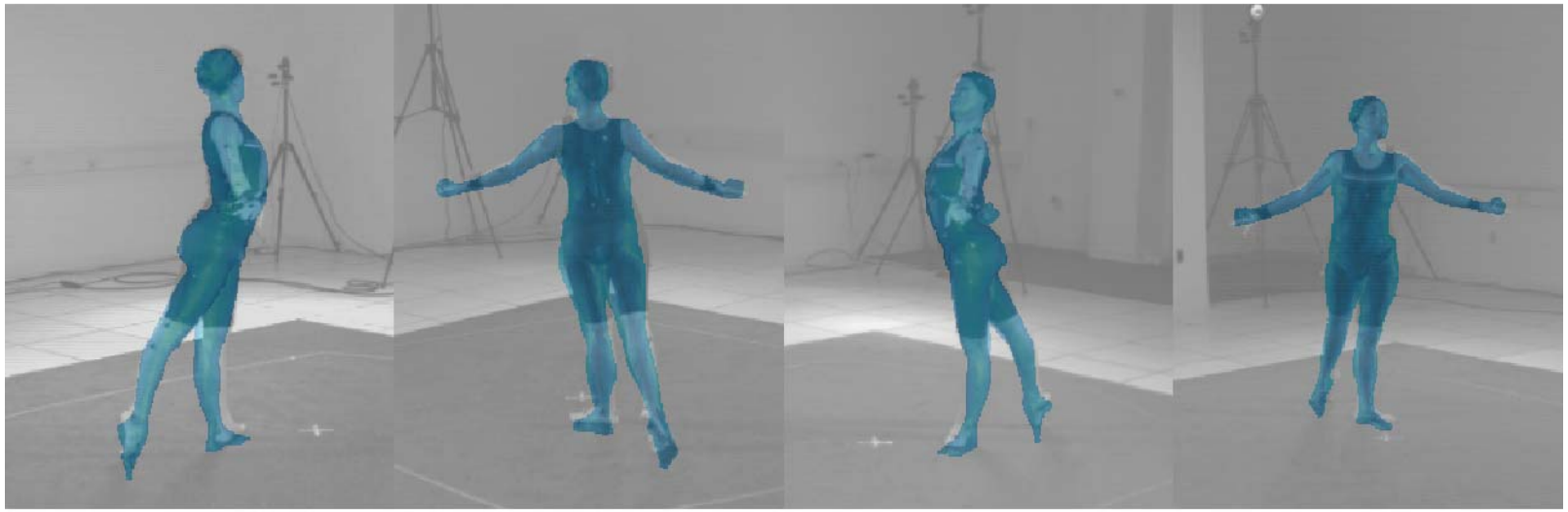
Optimization – Step by Step



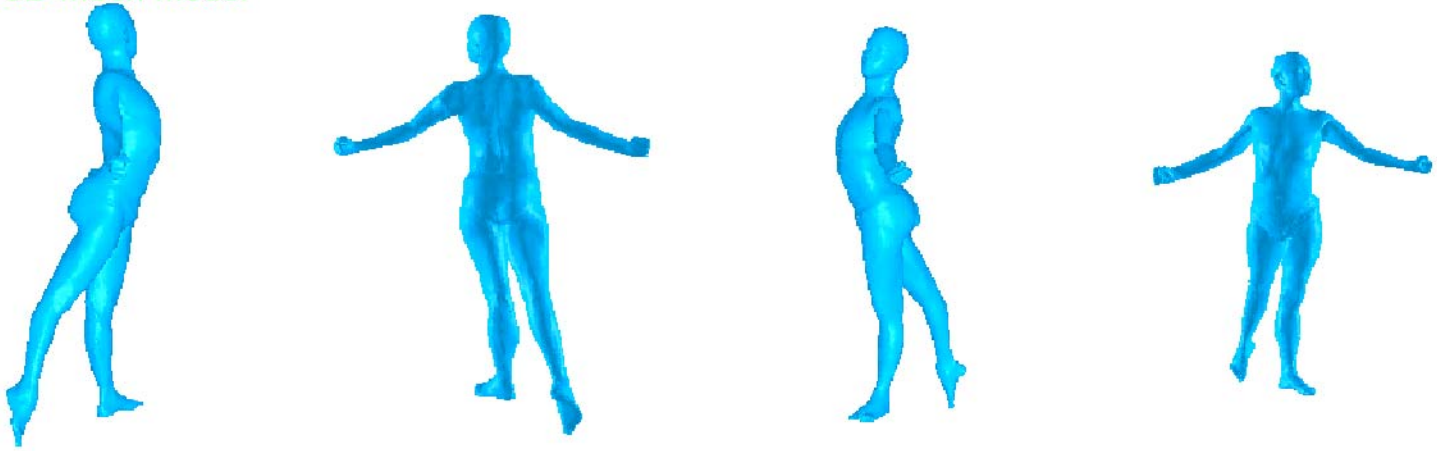
Optimization – Step by Step



Optimization – Step by Step



Fitted 3D mesh model



The “naked” case

20 bases.
Subject not in training set.

BROWN

Problem: Clothing

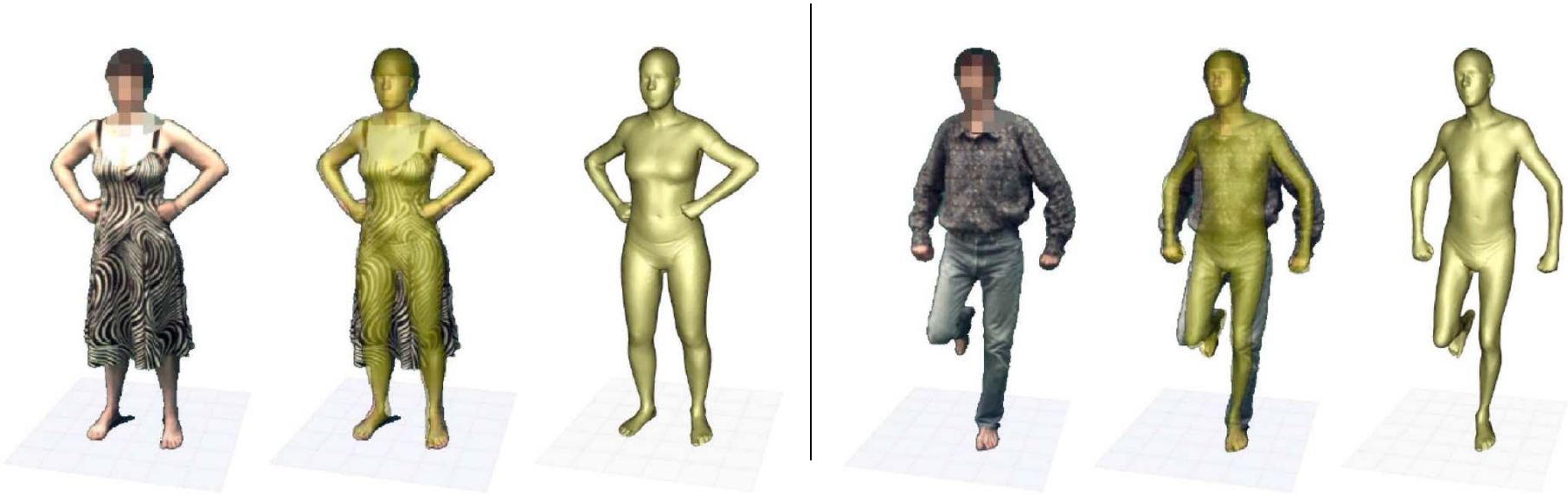
BROWN

Problem: Clothing

Shape under clothing

Can you guess what someone looks like under their clothes?

- Exploit the factorization of the model to combine constraints across poses.



Principle: Shape under clothing

- Silhouettes are larger when there is clothing

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes

$$E_{\text{inside}}(s) = d\left(\mathbf{F}^{e}_{k,s}, \mathbf{F}^{o}_{k}\right)$$

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes

$$E_{\text{inside}}(s) = d\left(\mathbf{F}^{e}_{k,s}, \mathbf{F}^{o}_{k}\right)$$

  - Should not try to explain the entire image silhouette

$$d\left(\mathbf{F}^{o}_{k}, \mathbf{F}^{e}_{k,s}\right)$$ **- NO**



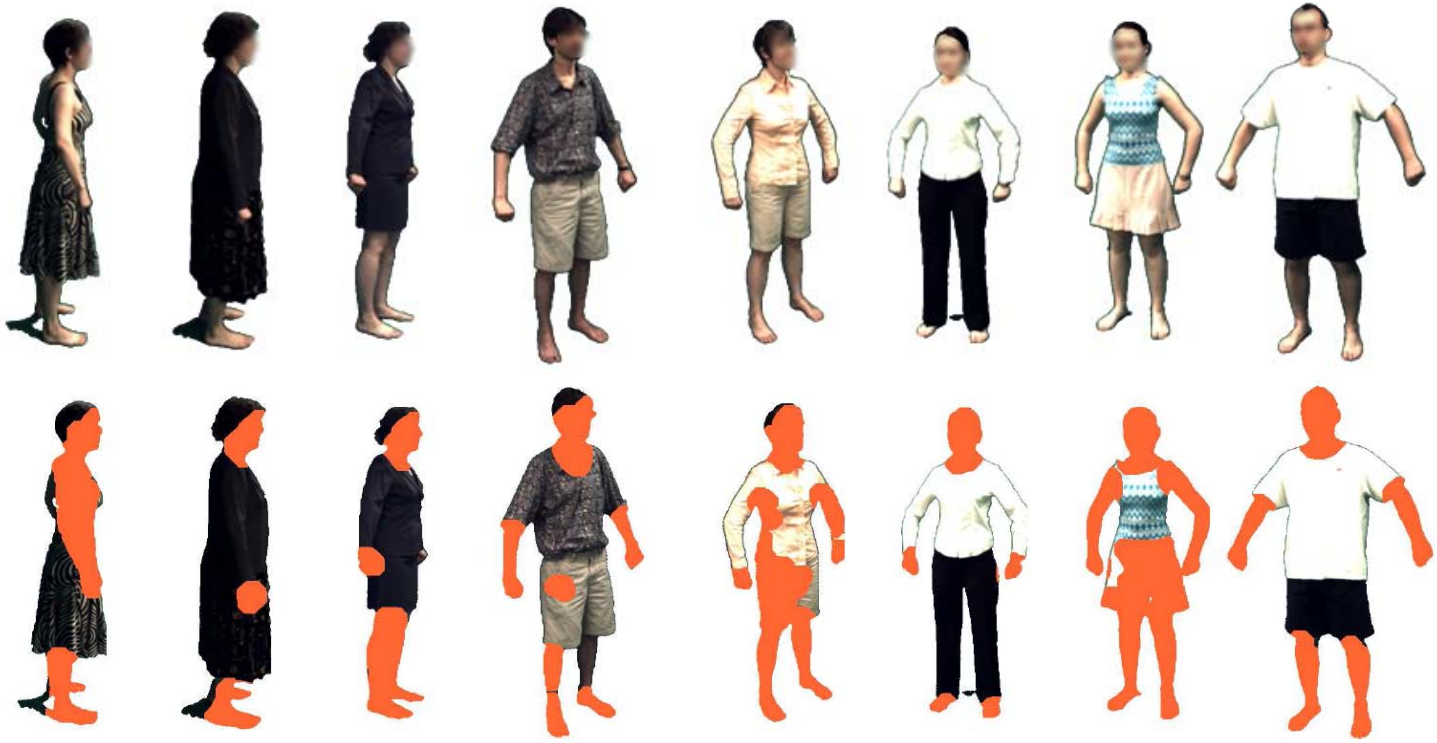

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight in regions without clothes

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight for skin regions

Skin Detection



Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight for skin regions

$$E_{\text{expand}}(s) = d\left(\mathrm{S}_k^o, \mathrm{F}_{k,s}^e\right) + \lambda d\left(\mathrm{F}_k^o \setminus \mathrm{S}_k^o, \mathrm{F}_{k,s}^e\right)$$

skin model $\lambda < 1$ non-skin model

Image

Skin / Non-skin Silhouettes

Model Silhouette

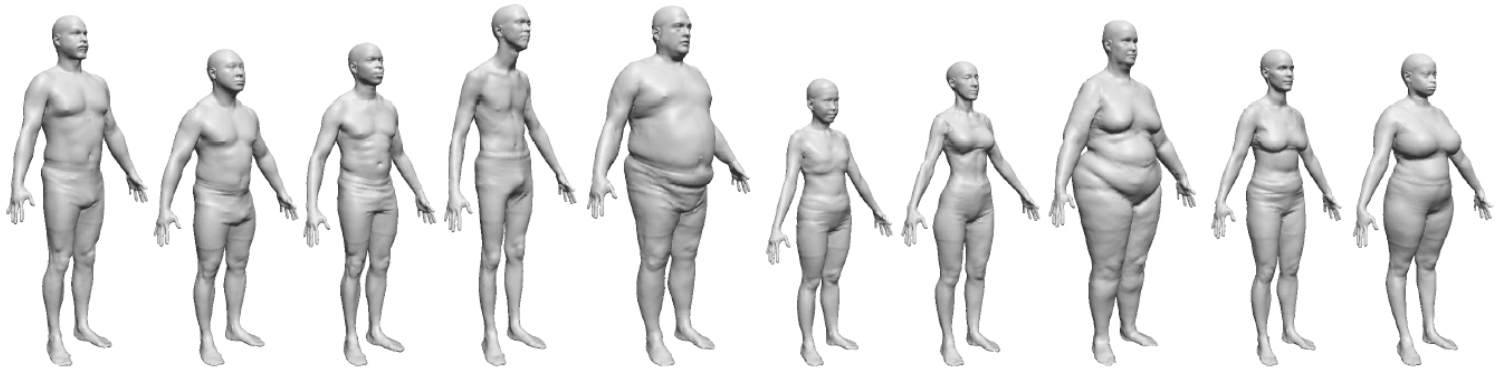
Overlap

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight for skin regions
- True shape not observable
  - Family of human body shapes (known statistics)

$$E_{shape}(\beta) = \sum_j \max\left(0, \frac{|\beta_j|}{\sigma_{\beta,j}} - \sigma_{thresh}\right)^2$$

$$E_{pose}(\theta)$$



[Allen et al. ’03]

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight for skin regions
- True shape not observable
  - Family of human body shapes (known statistics)

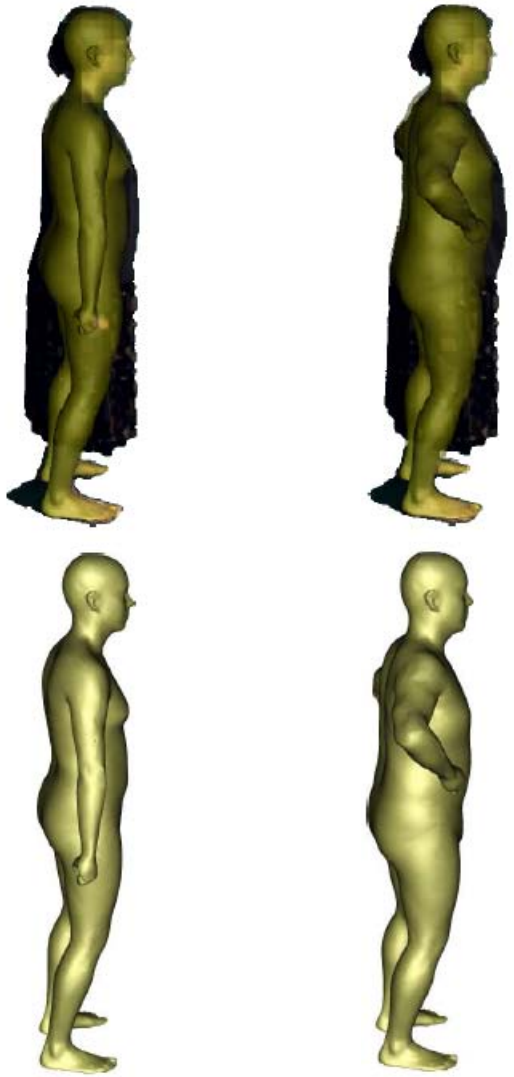
Objective Function

$$E_{\text{clothes}}(s) = \sum_{k=1}^{K} E_{\text{inside}}(s) + E_{\text{expand}}(s) + E_{\text{shape}}(\beta) + E_{\text{pose}}(\theta)$$

Comparison

Fitting as if Naked

Single-pose Fitting with Clothes



Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight for skin regions
- True shape not observable
  - Family of human body shapes (known statistics)
- **Body shape constant although pose may vary:**

Principle: Shape under clothing

- Silhouettes are larger when there is clothing
  - Body must fit inside silhouettes
  - Constraints are tight for skin regions
- True shape not observable
  - Family of human body shapes (known statistics)
- Combine constraints across pose

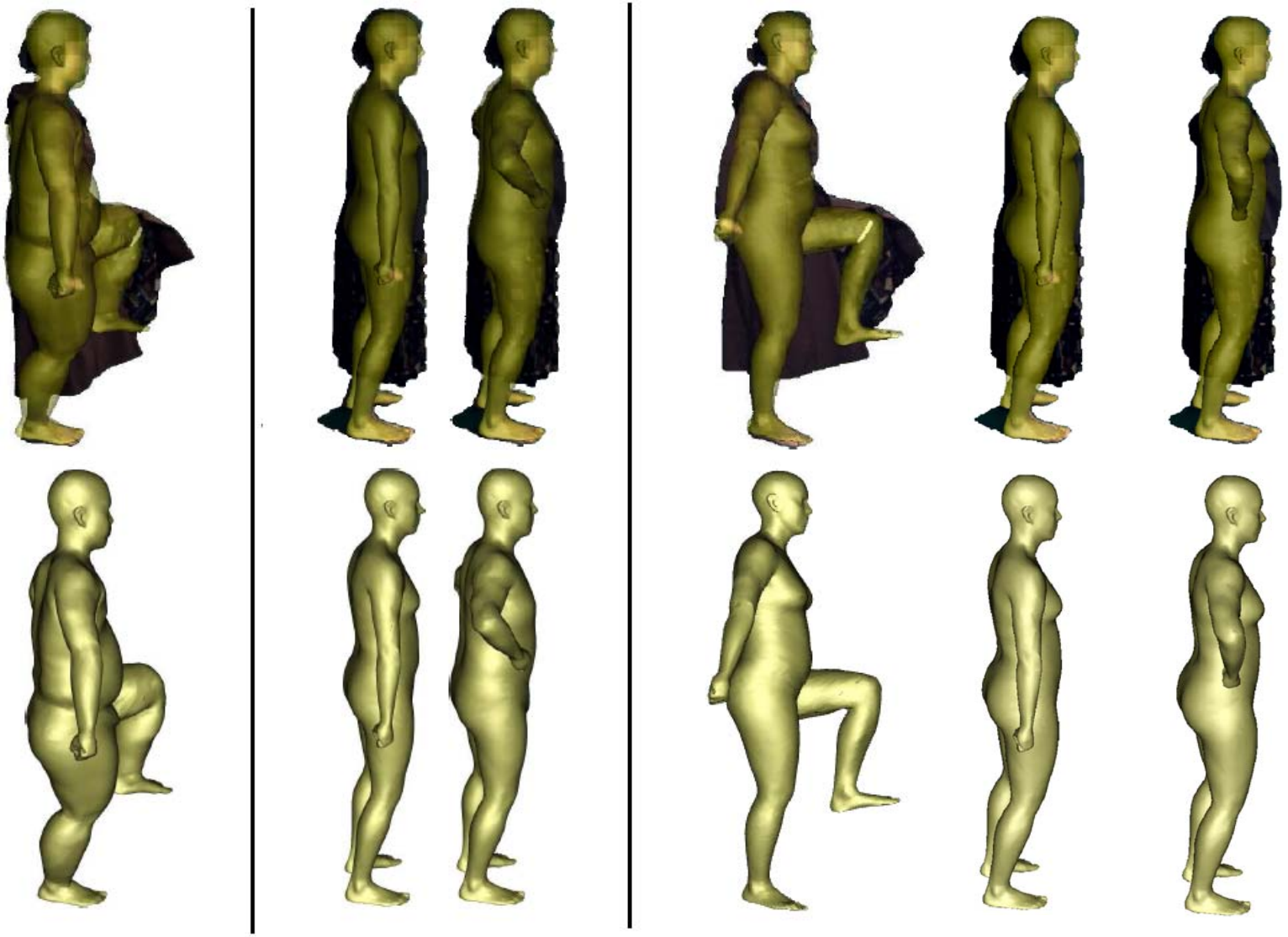
"Batch" objective function

$$E_{\text{clothes}}(\beta, \Theta) = \sum_{p=1}^{P} \sum_{k=1}^{K} E_{\text{inside}}(\beta, \theta_p) + E_{\text{expand}}(\beta, \theta_p) + E_{\text{shape}}(\beta) + E_{\text{pose}}(\theta_p)$$

$$\Theta = \theta_1, \ldots, \theta_P$$

Comparison

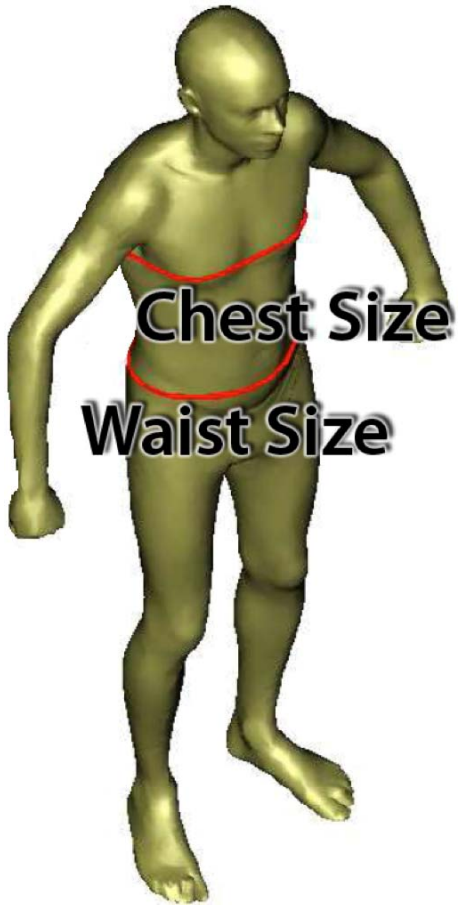
Fitting as if Naked | Single-pose Fitting | Batch Fitting



BROWN

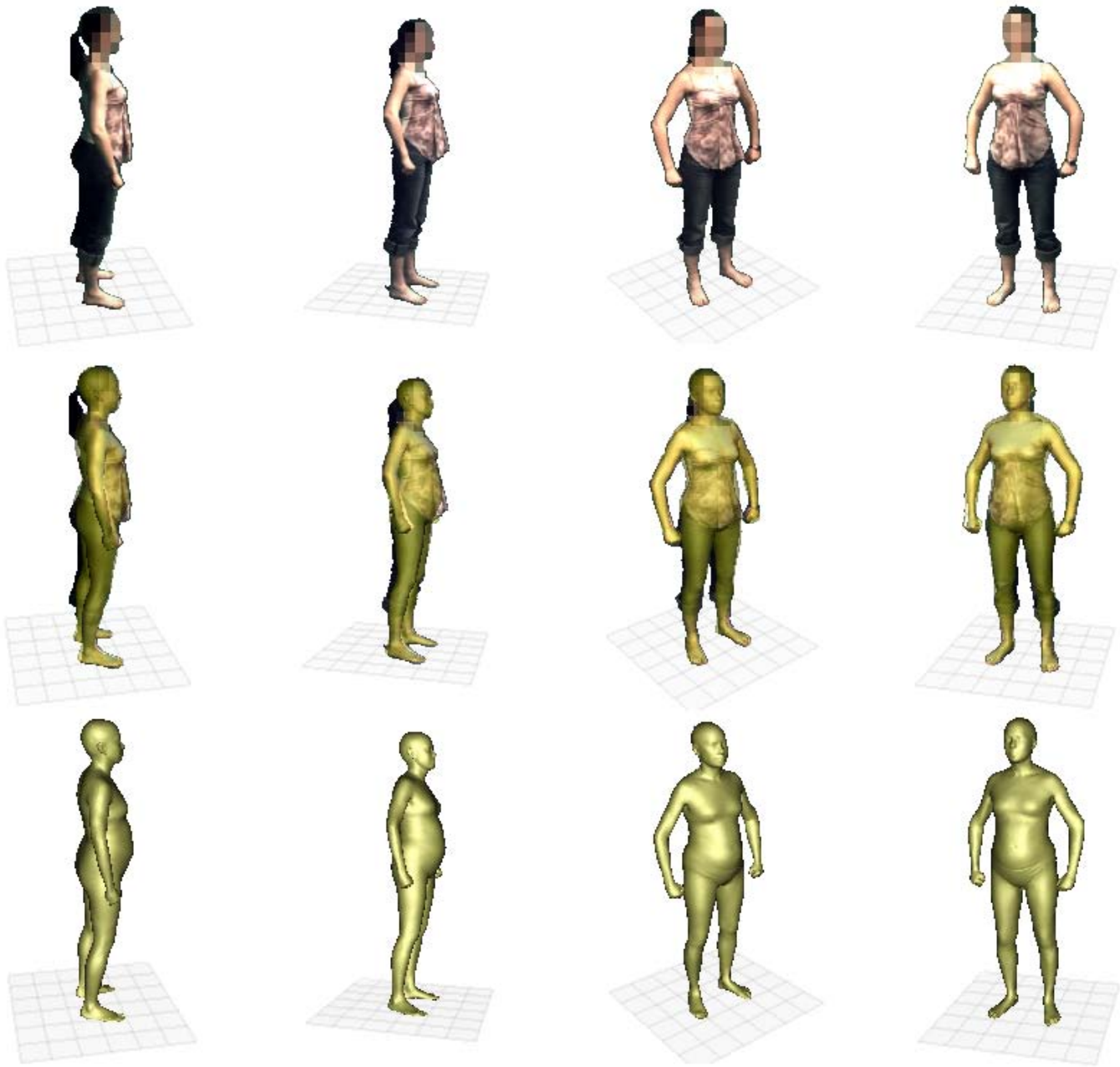
Shape under clothing

Quantitative Evaluation



BROWN

Failure case



BROWN

What about “regular” video?

- Everything here is multi-camera
  - See ICCV’07 for some monocular results (using cast shadows)
- Combine multiple poses to estimate shape
  - These poses can come from different cameras or different frames of the same camera.
    - Though pose estimation in a single frame is harder.
- There are many more cues to use beyond silhouettes
- Assume calibration here

BROWN

Applications

- Surveillance
  - Extract body shape measurements from surveillance video.
  - Gender from video (94.3% correct on our clothing dataset).
- Games
  - Avatar creation and animation.
- Fashion
  - Extract body shape measurements.
  - Virtual try-on.

BROWN

Summary of approach

- Estimating human shape may be as important as estimating 3D pose
- Detailed graphics model practical for vision
  - Sometimes complexity makes things easier (or possible).
  - Better match to image evidence (e.g. torso shape).
  - Initial optimization approx 5hrs (!) per frame; down to about 2min.
- Shape under clothing
  - Combine constraints on shape constancy and skin with clothing-appropriate observation model.
- Representation
  - Supports more than “detection”; is someone tall? fat? male? old?
  - 3D model allows us to explain illumination, shading, pose variation, muscle bulging, ….
  - This allow us to extract what is constant over time (identity, shape, etc.)

Current & future work

- Monocular estimation and tracking in multiple frames with moving camera
  - Constraints on shape consistency
- Beyond silhouettes
  - Internal structure & motion
- Modeling clothing and hair
- Dynamics of soft tissue and cloth
- Evaluation of accuracy
  - Pose and shape

See Balan *et al*, ICCV '07

The evolution of man

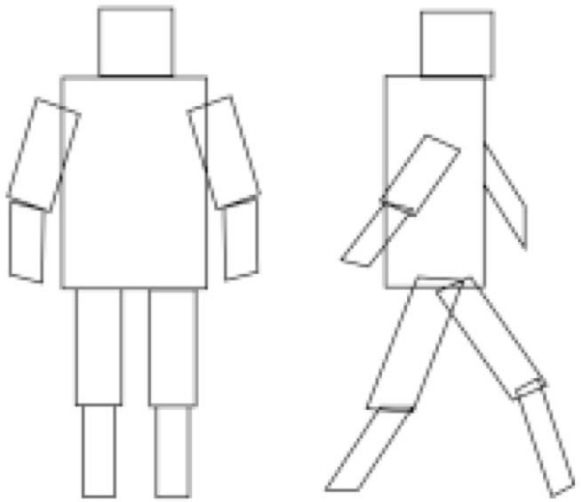
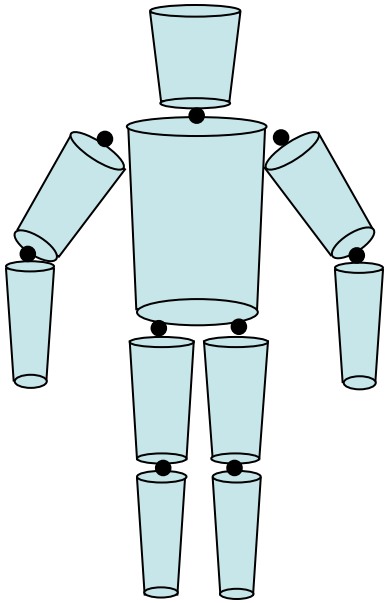


Figure 1: The cardboard person model. The limbs of a person are represented by planar patches.

1996



2000

2007

Today: Articulated Recognition

- Introduction / Challenges / Basic Approaches
- Regression approach
  - Urtasun and Darrell's local GP-model
- Pictorial Structures
  - Felzenswab and Huttenlocher
  - Ramanan et al.
- Strong local features
  - Lubomir Bourdev's poselets
- Strong global models
  - Black et al.'s work with the SCAPE model