Human Motion Analysis
Lecture 9: Image likelihood

Raquel Urtasun

TTI Chicago

May 6, 2010

Materials used for this lecture

- Slides about pictorial structures adapted from Daniel Huttenlocher's slides.
- See references when ever cited in the slides.

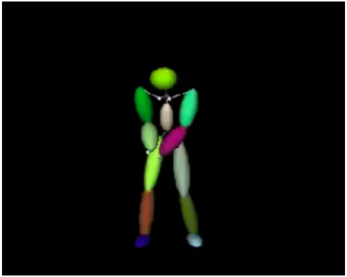
Contents of today's lecture?

We will look into generative approaches to pose estimation. We will focus on:

- image likelihoods

The problem of human pose estimation

- The goal is given an image $\mathbf{I}$ to estimate the 3D location and orientation of the body parts $\mathbf{y}$.



Pose estimation

- **Generative approaches:** focus on modeling

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

- **Discriminative approaches:** focus on modeling directly

$$p(\phi|\mathbf{I})$$

Today we will talk about generative approaches.
Later in the class we will cover discriminative approaches.

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Common to all of them is the need to model

- **Image likelihood:** $p(\mathbf{I}|\phi)$
- **Priors:** $p(\phi)$

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Common to all of them is the need to model

- **Image likelihood:** $p(\mathbf{I}|\phi)$
- **Priors:** $p(\phi)$

In general $p(\mathbf{I})$ is assumed constant and ignored. The different trackers then depend on the different modeling choices and optimization procedures.

Generative approaches

Generative approach models

$$p(\phi|\mathbf{I}) = \frac{p(\mathbf{I}|\phi)p(\phi)}{p(\mathbf{I})}$$

Types of generative approaches:

- **Bayesian approaches:** focus on approximating $p(\phi|\mathbf{I})$, usually via sampling (e.g., particle filter).
- **Optimization or energy-based techniques:** focus on computing the MAP or ML estimate of $p(\phi|\mathbf{I})$.

Common to all of them is the need to model

- **Image likelihood:** $p(\mathbf{I}|\phi)$
- **Priors:** $p(\phi)$

In general $p(\mathbf{I})$ is assumed constant and ignored. The different trackers then depend on the different modeling choices and optimization procedures.

In the next lectures we will look at ...

Priors: $p(\phi)$

- Joint limits
- Shape priors
- Pose priors
- Dynamical priors
- Physics

In this lecture we will look at ...

Likelihood models: $p(\mathbf{I}|\phi)$

- Monocular tracking: 2D-3D correspondences, silhouettes, edges, template matching, etc.
- Multi-view tracking: stereo, visual hull, etc.

Note that I have defined ϕ as a general quantity, not just the pose.

Monocular tracking

2D tracking

- Pictorial structures

3D tracking

- Silhouettes
- Skeleton
- Edges
- 2D to 3D correspondences
- Optical flow

Pictorial structures

- Local models of **appearance** with non-local geometric or **spatial** constraints
    - Image patches describing color, texture, etc
    - 2D spatial relations between pairs of patches
- Simultaneous use of appearance and spatial information since simple part models alone too non-distinctive

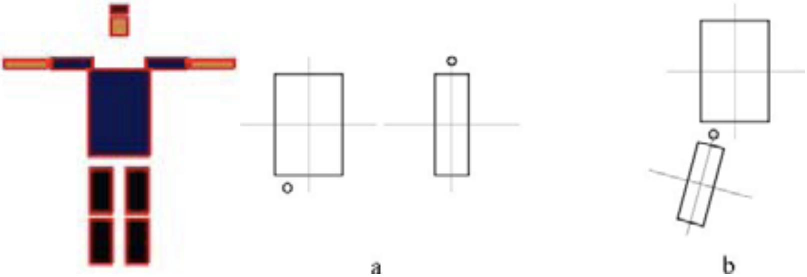


Figure: Pictorial structures (Felzenszwalb and Huttenlocher 04)

History of pictorial structures

- Pictorial structures date from early 1970s
- Practical recognition algorithms proved difficult.
- Purely geometric models widely used through early 1990s based on combinatorial matching to image features.
- Appearance based models also developed: Templates or patches of image, lose geometry.
- Other part-based models, but not seen in the class.

Definition of pictorial structures

The pictorial structure is represented by the following variables:

- Set of parts $V = \{v_1, \cdots, v_n\}$ and $\mathbf{L} = (\mathbf{l}_1, \cdots, \mathbf{l}_n)$ specifies the configuration of the parts.
- $\mathbf{A} = (\mathbf{a}_1, \cdots, \mathbf{a}_n)$ are appearance parameters.
- The relation between parts is a **Random field**.
- The edges $e_{i,j} \in \mathcal{E}$ represent the connexion between different neighboring parts, which express the explicit dependencies.
- The connection parameters $\mathbf{C} = \{\mathbf{c}_{i,j} | \; \forall e_{i,j} \in \mathcal{E}\}$

Learning and Inference in pictorial structures

- The model is defined as $\mathcal{M} = (\mathbf{A}, \mathbf{E}, \mathbf{C})$.
- **Learning** the model $\mathcal{M}$ is performed from labeled example images $\mathbf{I}_1, \cdots, \mathbf{I}_m$ and configurations $\mathbf{L}_1, \cdots, \mathbf{L}_m$.
- Typically a parametric form of $\mathbf{A}$ and $\mathbf{C}$ is employed.
  - e.g., $\mathbf{a}_i$ constant color rectangle: learn the average color and variation.
  - e.g., $\mathbf{c}_{i,j}$: relative translation of parts: learn the average position and variation.
- **Inference**: Find most likely location $\mathbf{L}$ for the parts in $\mathbf{I}$, or multiple highly likely locations.
- Inference is done by evaluating the image likelihood: how likely it is that model is present.
- The state is $\phi = \mathbf{L}$.

Standard Bayesian approach

- The state is $\phi = \mathbf{L}$ and the model $\mathcal{M} = (\mathbf{A}, \mathbf{E}, \mathbf{C})$.
- Estimate posterior distribution $p(\phi|\mathbf{I}, \mathcal{M})$.
- Find maximum (MAP) or high values (sampling).
- Generative tracking

$$p(\phi|\mathbf{I}, \mathcal{M}) \propto p(\mathbf{I}|\phi, \mathcal{M}) p(\phi|\mathcal{M})$$

which is composed of likelihood $p(\mathbf{I}|\phi, \mathcal{M})$ and the prior $p(\phi|\mathcal{M})$.

Class of Models I

- The computational difficulty depends on the posterior distribution.
- One can exploit the structure of the graph $G = (\mathbf{V}, \mathbf{E})$ which represents a **Markov Random Field (MRF)**, each node explicitly depends on its neighbors.
- If G is a **tree**:
  - Natural for models of animate skeletons
  - Prior can be computed efficiently
  - Prior on relative location

$$p(\phi|\mathbf{E}, \mathbf{C}) = \prod_{E} p(\mathbf{l}_i, \mathbf{l}_j | \mathbf{c}_{i,j})$$

Class of Models II

- Image likelihood is usually the product of individual likelihoods

$$p(\mathbf{I}|\phi, \mathcal{M}) = \prod_i p(\mathbf{I}|\mathbf{l}_i, \mathbf{a}_i)$$

- Good approximation when parts dont overlap.
- The form of connections is also important: space with deformation distance

$$p(\mathbf{l}_i, \mathbf{l}_j|c_{i,j}) = \mathcal{N}(T_{i,j}(\mathbf{l}_i) - T_{j,i}(\mathbf{l}_i), |0, \Sigma_{i,j})$$

  is a normal distribution in a transformed space
- $T_{i,j}$ and $T_{j,i}$ capture ideal relative locations of parts and $\Sigma_{i,j}$ measures deformation.
- It's the Mahalanobis distance in transformed space (weighted squared Euclidean distance).

Bayesian formulation of learning

- Supervised learning: we are given example images $\mathbf{I}_1, \cdots, \mathbf{I}_m$ with configurations $\mathbf{L}_1, \cdots, \mathbf{L}_m$.
- Obtain estimates of the model given i.i.d. samples

$$\max_{\mathcal{M}} p(\mathbf{I}_1, \cdots, \mathbf{I}_m, \mathbf{L}_1, \cdots, \mathbf{L}_m | \mathcal{M}) = \prod_k p(\mathbf{I}_k, \mathbf{L}_k | \mathcal{M})$$

- Rewrite joint probability as product of appearance and dependencies separate

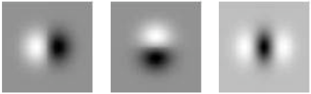
$$\max_{\mathcal{M}} \prod_k p(\mathbf{I}_k | \mathbf{L}_k, \mathbf{A}) \prod_k p(\mathbf{L}_k | \mathbf{E}, \mathbf{C})$$

Learning models efficiently

- Estimating appearance $p(\mathbf{I}_k|\mathbf{L}_k, \mathbf{A})$ is typically done by ML estimation
- E.g., for constant color patch use Gaussian model, computing mean color and covariance
- Estimating dependencies $p(\mathbf{L}_k|\mathbf{E}, \mathbf{C})$
  - Estimate **C** for pairwise locations $p(\mathbf{l}_{i,k}, \mathbf{l}_{j,k}|\mathbf{c}_{i,j})$.
  - E.g., for translation compute mean offset between parts and variation in offset.
  - Best tree using **minimum spanning tree (MST) algorithm**. It computes the pairs with smallest relative spatial variation

Example: Generic face model

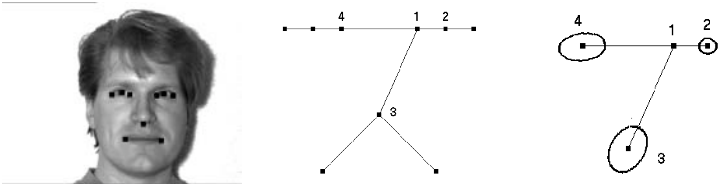
- Each part $\mathbf{a}_i$ is a local image patch represented as response to oriented filters



- Pairs of parts constrained in terms of their relative (x, y) position in the image.
- Consider two models: 5 parts and 9 parts
    - 5 parts: eyes, tip of nose, corners of mouth
    - 9 parts: eye split into pupil, left side, right side

Learned face model

- Appearance and structure parameters learned from labeled frontal views.
- Structure captures pairs with most predictable relative location least uncertainty
- Gaussian (covariance) model captures direction of spatial variations differs per part



Example: Generic Person Model

- Each part represented as rectangle with fixed width, varying length: Learn average and variation.
- Connections approximate revolute joints: joint location, relative position, orientation, foreshortening.
- Learned 10 part model: All parameters learned including joint locations

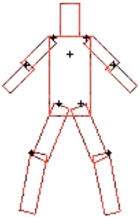


Figure: Pictorial structures learned for a human (Felzenszwalb and Huttenlocher 04)

Bayesian formulation of recognition I

- Given model $\mathcal{M}$ and image $\mathbf{I}$, seek good configuration $\mathbf{L}$.
- This can be done by MAP estimation $\max_{\mathbf{L}} p(\mathbf{L}|\mathbf{I}, \mathcal{M})$ or by sampling.
- Brute force solutions intractable: With n parts and s possible discrete locations per part, $\mathcal{O}(s^n)$.

Bayesian formulation of recognition II

- However, we can use the graph structure (MRF) such that

$$\max_{\mathbf{L}} p(\mathbf{L}|\mathbf{I}, \mathcal{M}) = \max_{\mathbf{L}} \prod_{v} p(\mathbf{I}|\mathbf{l}_i, \mathbf{a}_i) \prod_{E} p(\mathbf{l}_i, \mathbf{l}_j | \mathbf{c}_{i,j})$$

- Taking logarithms we have

$$\min_{\mathbf{L}} -\log p(\mathbf{L}|\mathbf{I}, \mathcal{M}) = \min_{\mathbf{L}} \sum_{v} m_j(\mathbf{l}_j) + \sum_{E} d_{i,j}(\mathbf{l}_i, \mathbf{l}_j)$$

- Typically dynamic programming is used to solve this efficiently by recursively computing

$$B_j(\mathbf{l}_i) = \min_{\mathbf{l}_j} \left(m_j(\mathbf{l}_j) + d_{i,j}(\mathbf{l}_i, \mathbf{l}_j) + \sum_{C_j} B_c(\mathbf{l}_j) \right)$$

  where C_j are the children of node j

- The running time is now $\mathcal{O}(ns^2)$ for n parts and s locations.

Recognizing Faces

- Generic model of frontal view
  - Using learned 5- and 9-part models
  - Local oriented filters for parts
  - Relatively small spatial variation in part locations
  - Similar overall size and orientation of face
- MAP estimation to find best match
  - Posterior estimate of configuration **L** is accurate because parts do not overlap
  - Consider all possible locations in image
  - Very efficient: runs in real time

Examples of detections

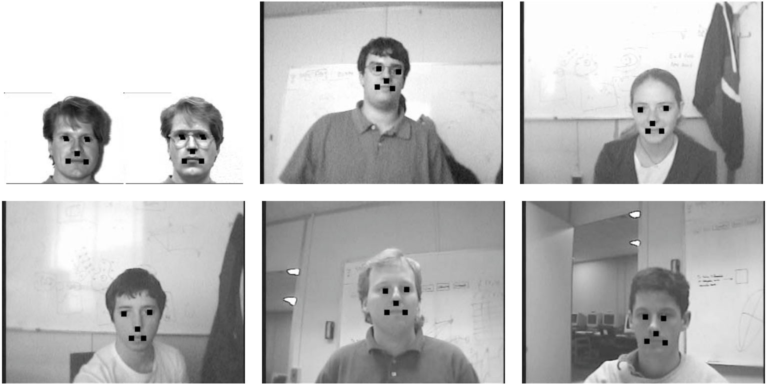


Figure: Examples of detected faces (Felzenszwalb and Huttenlocher 04)

Recognizing People

- Frontal view models
    - Generic model using binary rectangles for parts match to "difference image".
    - Specific model using color rectangles for parts: match to original image.
- Sampling posterior to find good matches: posterior estimate of **L** can be high for several configurations due to overlap of parts.
    - Generate good possible matches as hypotheses:locations with $p(\mathbf{L}|\mathbf{I}, \mathcal{M})$ is large.
    - Validate using another technique: here using Chamfer distance, a correlation like measure.
    - Use best of 200 samples search over all locations runs in under minute.

Samples from the posterior

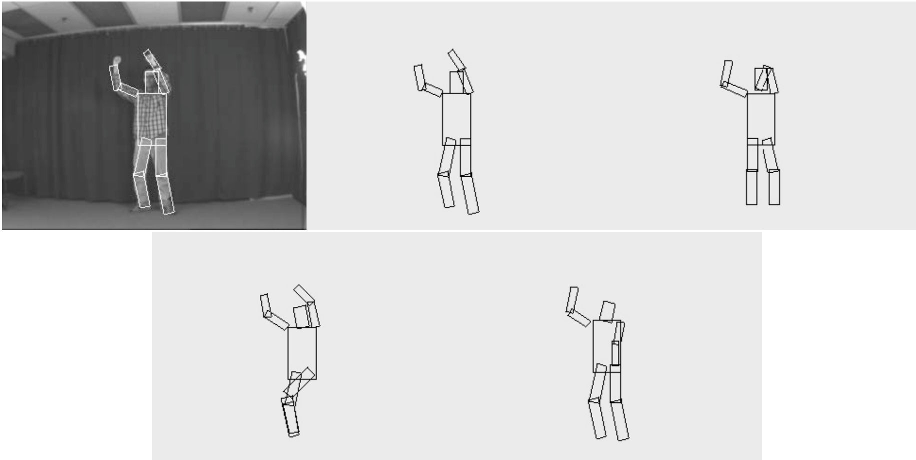


Figure: Examples of posterior samples (Felzenszwalb and Huttenlocher 04)

Recognizing people with clutter

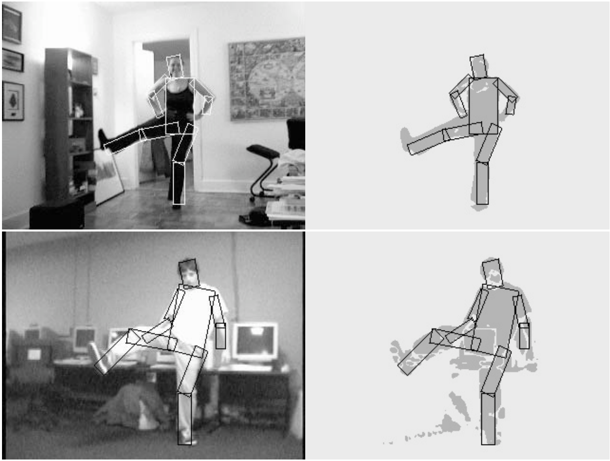


Figure: Examples of detected humans (Felzenszwalb and Huttenlocher 04)

Recognizing a variety of poses

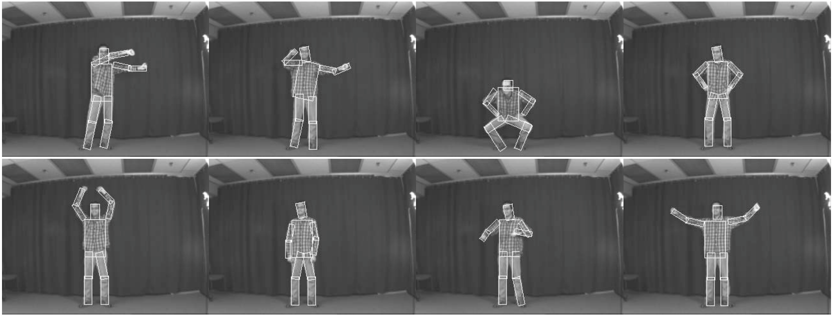


Figure: Examples of detected poses (Felzenszwalb and Huttenlocher 04)

Model of specific person

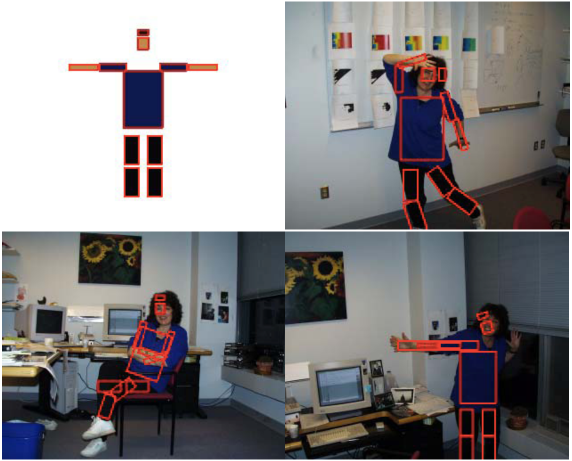


Figure: Examples of detected humans (Felzenszwalb and Huttenlocher 04)

Extensions of pictorial structures

- (Ramanan 06) model them with Conditional Random Fields (CRFs), casting of visual inference as an iterative parsing process, where one sequentially learns better and better features tuned to a particular image.
- Hallucinate occlusions

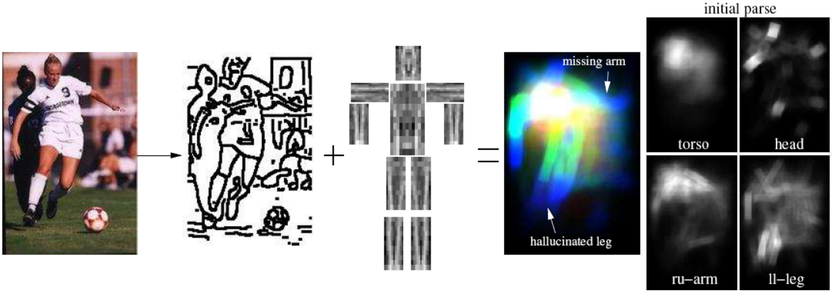


Figure: Pictorial structures with CRFs (Ramanan 06)

Learning appearance model

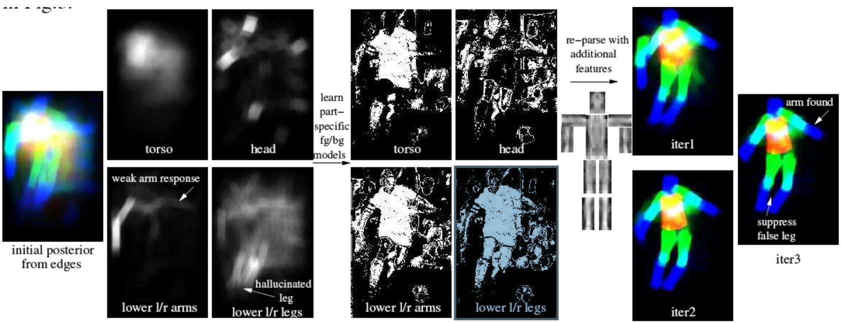
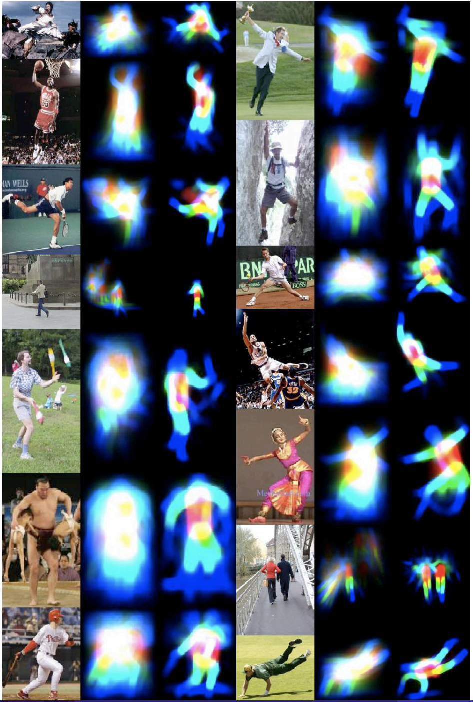


Figure: Learning pictorial structures (Ramanan 06)



Monocular tracking

2D tracking

- Pictorial structures

3D tracking

- Silhouettes
- Skeleton
- Edges
- 2D to 3D correspondences
- Optical flow

For 3D tracking we represent the likelihood in terms of error functions

$$-\log p(\mathbf{I}|\phi) = E$$

with E a combination of error functions

Monocular tracking

2D tracking

- Pictorial structures

3D tracking

- Silhouettes
- Skeleton
- Edges
- 2D to 3D correspondences
- Optical flow

For 3D tracking we represent the likelihood in terms of error functions

$$-\log p(\mathbf{I}|\phi) = E$$

with E a combination of error functions

Silhouettes: area of overlap

- Silhouettes are typically obtained from background substraction
- Two types of likelihood function
  - Area of overlap
  - Fit the inside of the silhouette: distance transform

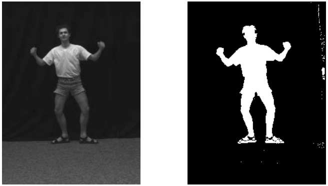


Figure: Silhouettes (Sminchisescu et al 02)

Area of overlap

- Maximize the model to image silhouette area of overlap

$$E_{align} = \frac{1}{2\sigma_{alig}^2} f(\sum_{t \in V_t} (S_a - S_g)^2)$$

where S_g is the area of the target silhouette, and S_a is the area of the silhouette of the projected surface. f since otherwise we would like to maximize.

Silhouettes and distance transform

- Pushes the model inside the image silhouette

$$E_{dist} = \frac{1}{2\sigma_{dist}^2} \sum_i e_{s_i}(r_i(x), S_g)$$

where i ranges over all projected model nodes, and e_{s_i} is the distance from a predicted model point $r_i(x)$ to a given silhouette S_g.

- e_{s_i} can be estimated by computing the distance transform D of the silhouette S_g and evaluating it in the points i

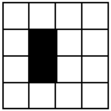
$$e_{s_i}(r_i(x), S_g) = D(r_i(x))$$

Distance transform

- One typical example is to define

$$d(\mathbf{x}, \mathbf{P}) = \min_{y \in \mathbf{P}} ||\mathbf{x} - \mathbf{y}||_2^2$$

where P is a set of points.



1	1	1	2
1	0	1	2
1	0	1	2
1	1	1	2

Figure: Distance transform from silhouettes (Felzenszwalb et al 04)

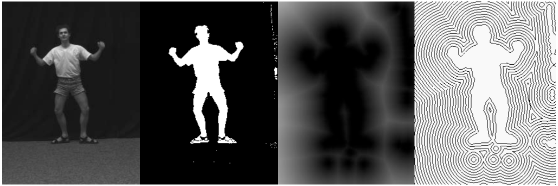


Figure: Distance transform from silhouettes (Sminchisescu et al 02)

Influence of both silhouette terms

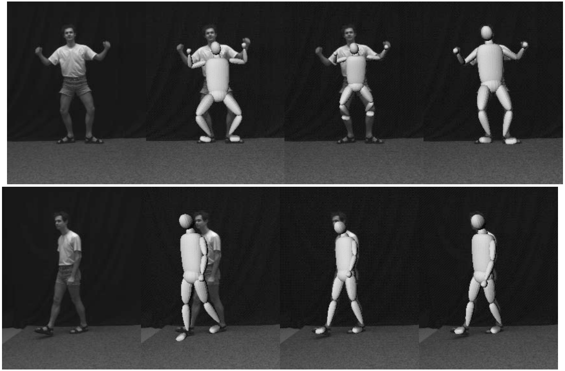


Figure: Model estimation based on various silhouette terms original images (a,e), initial models (b,f), silhouette attraction term only (c,g), silhouette attraction and area overlap terms (d,h)(Sminchisescu et al 02)

Skeleton

- Represent directly the projection of the skeleton into the image by evaluating the new distance transform.

$$E_{skel} = \frac{1}{2\sigma_{skel}^2} \sum_i D(r_i(x))$$

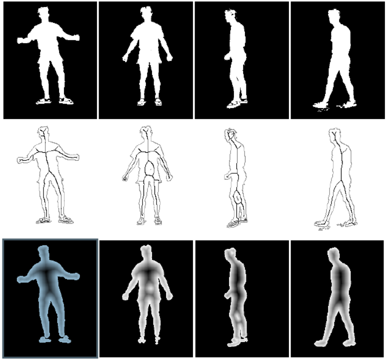


Figure: Skeleton representation (Sminchisescu et al 02)

Edges

- Minimize the distance of the projected edges to the image edges.
- Do the search incrementally

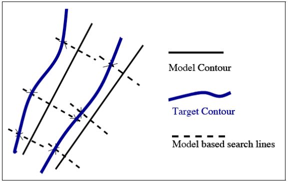


Figure: Edge search (Sminchisescu et al 02)

- More robust to miss-alignements by using a distance transform

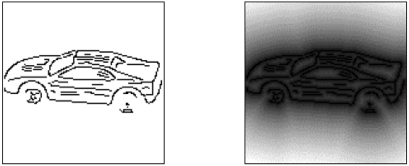


Figure: Edge distance transform (Felzenszwalb et al 04)

2D to 3D correspondences

- Minimize the distance between the projection of the 3D model and the tracked 2D points.

$$E_{2D} = \frac{1}{2\sigma_{2D}^2} \sum_{j=1}^{J} ||\mathbf{m}_j - P(p_j(\phi))||_2^2$$

with m_j the j-th 2D tracked point, and $P(\mathbf{p}_j(\phi))$ the projection of a 3D point $\mathbf{p}_j$ which is a function of the state ϕ.

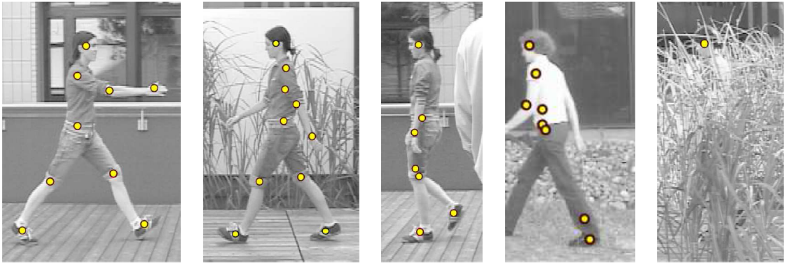


Figure: 2D to 3D correspondences (Urtasun et al. 06)

An alternative error function

- An alternative parameterization is in 3D using the line of sight: Plucker lines
- This can be used for 2D to 3D correspondences or for silhouettes

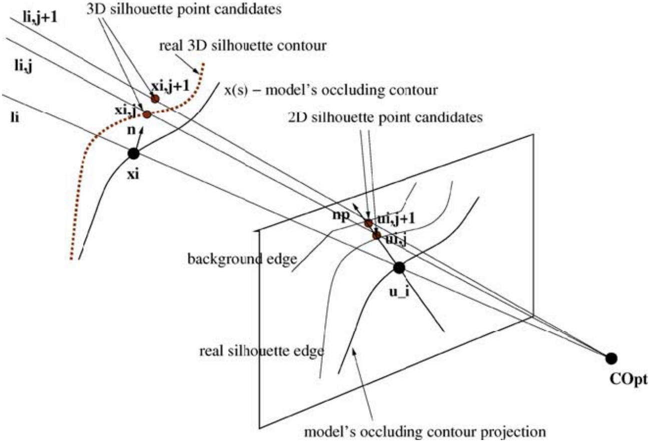

li,j+1
3D silhouette point candidates
real 3D silhouette contour
li,j
xi,j+1
x(s) – model's occluding contour
xi,j
li
n
2D silhouette point candidates
xi
np
ui,j+1
ui,j
background edge
u_i
real silhouette edge
COpt
model's occluding contour projection

Figure: 2D to 3D correspondences and edges (Ilic et al. 07)

Optical flow I

- **Optical flow** is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between the camera and the scene.
- Optical flow methods try to calculate the motion between two image frames which are taken at times t and $t + \delta t$ at every voxel position
- Assuming small movements and doing a Taylor expansion of first order

$$\mathbf{I}(x + \delta x, y + \delta y, t + \delta_t) \approx \mathbf{I}(x, y, t) + \frac{\partial \mathbf{I}}{\partial x}\partial x + \frac{\partial \mathbf{I}}{\partial y}\partial y + \frac{\partial \mathbf{I}}{\partial t}\partial t$$

- From these equations it follows that

$$\frac{\partial \mathbf{I}}{\partial x} v_x + \frac{\partial \mathbf{I}}{\partial y} v_y + \frac{\partial \mathbf{I}}{\partial t} = 0$$

  with $v_x = \frac{\delta x}{\delta t}$ and $v_y = \frac{\delta y}{\delta t}$ the components of the optical flow.
- This is usually written as

$$\nabla \mathbf{I}^T \mathbf{v} = -\mathbf{I}_t$$

Optical flow II

- Build 2D to 3D correspondences between consecutive frames

$$E_{flow} = \frac{1}{2\sigma_{flow}^2} \sum_i ||\mathbf{v}_i - \mathbf{d}(\phi)||_2^2$$

where $\mathbf{v}_i$ is an estimate of the flow, and $\mathbf{d}$ relates the point in the model at the previous instance with the new time instance.

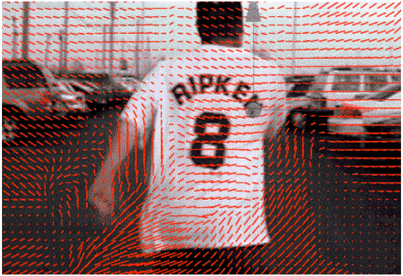


Figure: Optical flow

Multiview tracking

- Monocular likelihoods independent for every camera
- Stereo
- Shape from silhouettes
- 3D to 3D correspondences
- Shape from shadows

Stereo

- Stereo: shape from motion between two views
- It requires camera calibration for the internal parameters and correspondences

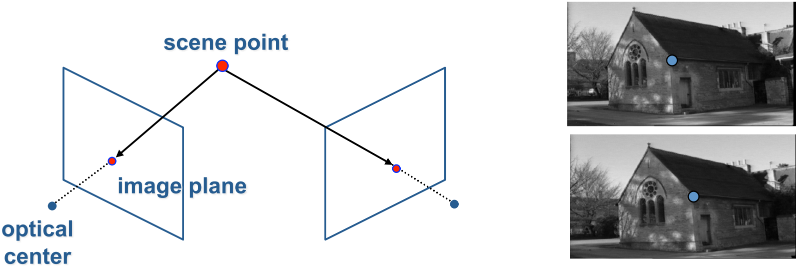


Figure: Estimation depth with stereo (Grauman)

Stereo likelihood

- The stereo reconstruction error can be computed as

$$E_{stereo} = \frac{1}{2\sigma^2_{stereo}} dist(\mathcal{M}, \mathbf{S})$$

where **S** is the stereo cloud and $\mathcal{M}$ is the 3D model.

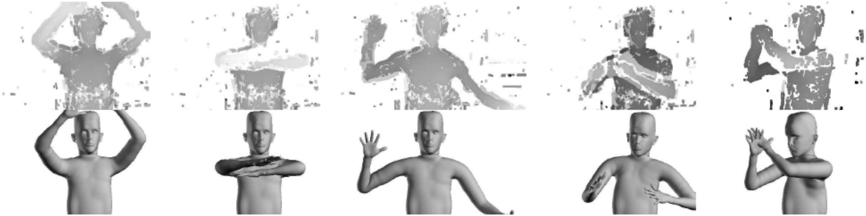


Figure: Skeleton fitting to stereo data (Plaenkers et al 03)

Shape from silhouettes

- The **visual hull** is the volume created by shape-from-silhouette 3D reconstruction.
- It assumes the foreground object in an image can be separated from the background, and segmented into a silhouette.
- The silhouette defines a back-projected generalized cone that contains the actual object. This cone is called a silhouette cone.

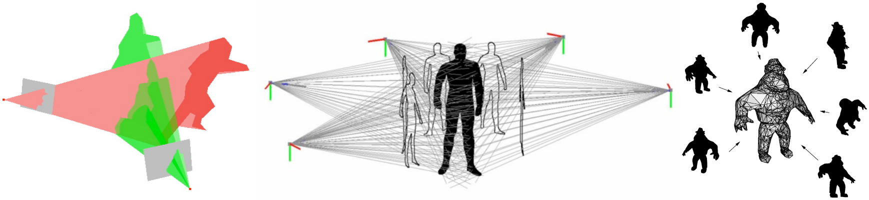


Figure: Visual hull

Shape from silhouettes

- The **visual hull** is the volume created by shape-from-silhouette 3D reconstruction.
- It assumes the foreground object in an image can be separated from the background, and segmented into a silhouette.
- The silhouette defines a back-projected generalized cone that contains the actual object. This cone is called a silhouette cone.
- The visual hull error can be computed as

$$E_{hull} = \frac{1}{2\sigma_{hull}^2} dist(\mathcal{M}, \mathbf{H})$$

with $\mathcal{M}$ the shape representation of the 3D model and $\mathbf{H}$ the visual hull.

Problems of Shape from silhouettes

- Require a 3D reconstruction step → time consuming
- Fail when silhouette information is used with only few cameras

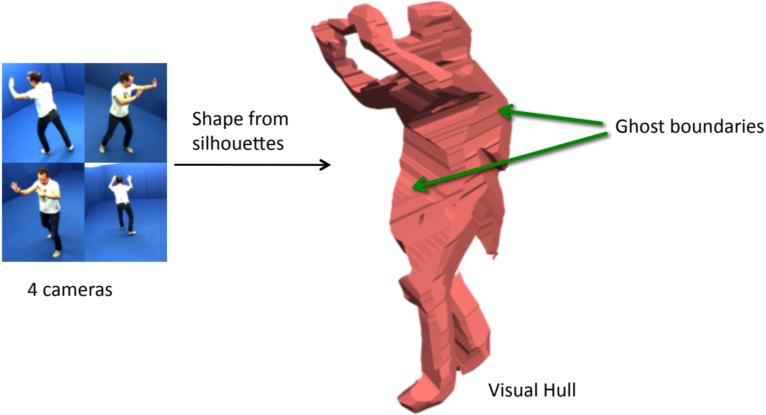


Figure: Ballan et al 08

3D to 3D correspondences

- The error function will simply be

$$E_{3D} = \frac{1}{2\sigma_{3D}^2} \sum_{i}^{M} ||\mathbf{m}_i - \mathbf{p}_i(\phi)||_2^2$$

where $\mathbf{m}_i$ and $\mathbf{p}_i$ are two points in correspondence.

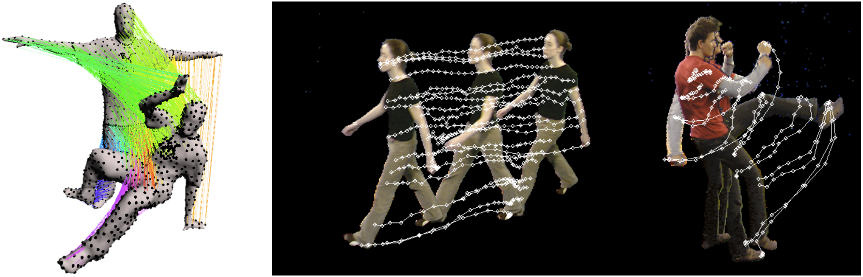


Figure: 3D to 3D correspondences (Stark and Hilton 05 and 07)

Examples of 3D to 3D correspondences

Shape from shadows

- Create an additional camera by detecting the shadow under strong illumination conditions

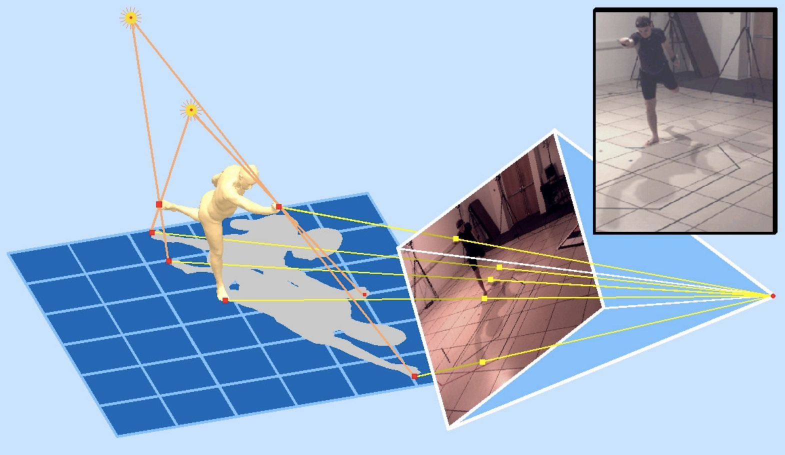


Figure: 3D from shadows (Balan et al 07)

Some impressive tracking results

Figure: Ballan et al 08

Some impressive tracking results

Figure: Ballan et al 08

Some impressive tracking results

Figure: Ballan et al 08

Some impressive tracking results

Figure: Ballan et al 08

Some impressive tracking results

Figure: Ballan et al 08

Some impressive tracking results

Figure: Ballan et al 08

More?

- Multi-view tracking in control environments is more or less solve
- More complex interactions between multiple subjects
- Outdoor environments are still challenging
- Monocular tracking is unsolved
- If you want to learn more, look at the additional material.
- Otherwise, do the research project on this topic!
- Next week we will look into physical priors