Computer Vision II - Lecture 14

Articulated Tracking II

10.07.2014

Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de
leibe@vision.rwth-aachen.de

Outline of This Lecture

- Single-Object Tracking
- Bayesian Filtering
  - Kalman Filters, EKF
  - Particle Filters
- Multi-Object Tracking
  - Data association
  - MHT, (JPDAF, MCMCDA)
  - Network flow optimization
- Articulated Tracking
  - GP body pose estimation
  - (Model-based tracking, AAMs)
  - Pictorial Structures

Topics of This Lecture

- Articulated Tracking
  - Motivation
  - Classes of Approaches
- Body Pose Estimation as High-Dimensional Regression
  - Representations
  - Training data generation
  - Latent variable space
  - Learning a mapping between pose and appearance
- Review: Gaussian Processes
  - Formulation
  - GP Prediction
  - Algorithm
- Applications
  - Articulated Tracking under Egomotion

Basic Classes of Approaches

- Global methods
  - Entire body configuration is treated as a point in some high-dimensional space.
  - Observations are also global feature vectors.
  - View of pose estimation as a high-dimensional regression problem.
  - Often in a subspace of “typical” motions...
- Part-based methods
  - Body configuration is modeled as an assembly of movable parts with kinematic constraints.
  - Local search for part configurations that provide a good explanation for the observed appearance under the kinematic constraints.
  - View of pose estimation as probabilistic inference in a dynamic Graphical Model.

Recap: Advantage of Silhouette Data

- Synthetic training data generation possible!
  - Create sequences of „Pose + Silhouette” pairs
  - Poses recorded with Mocap, used to animate 3D model
  - Silhouette via 3D rendering pipeline

Recap: Latent Variable Models

- Joint angle pose space is huge!
  - Only a small portion contains valid body poses.
  - Restrict estimation to the subspace of valid poses for the task
  - Latent variable models: PCA, FA, GPLVM, etc.
Recap: Articulated Motion in Latent Space

- Regression from latent space to:
  - Pose: \( p(\text{pose} \mid z) \)
  - Silhouette: \( p(\text{silhouette} \mid z) \)
- Regressors need to be learned from training data.

Recap: Learning a Generative Mapping

- Body Pose
  - Learn dim. red. (LLE)
  - X: Body Pose (high dim.)
  - Project pose
  - Y: Appearance Descriptor (low dim.)
- Appearance

Recap: Gaussian Process Regression

- "Regular" regression: \( y = f(x) \)
- GP regression: \( p(y(x)) \sim \mathcal{N}(\mu(x), \sigma(x)) \)

Recap: GP Prediction w/ Noisy Observations

- Calculation of posterior:
  - Corresponds to conditioning the joint Gaussian prior distribution on the observations:
  \[
  f_0 | X, X_0, t \sim \mathcal{N}(\hat{f}_0, \text{cov}(\hat{f}_0))
  \]
  - With:
  \[
  \hat{f}_0 = K(X_0, X) (K(X, X) + \sigma_f^2\mathbf{I})^{-1} t
  \]
  \[
  \text{cov}(\hat{f}_0) = K(X_0, X) - K(X_0, X) (K(X, X) + \sigma_f^2\mathbf{I})^{-1} K(X, X_0)
  \]
  - This is the key result that defines Gaussian process regression!

Recap: Articulated Multi-Person Tracking

- Idea: Only perform articulated tracking where it’s easy!
- Multi-person tracking:
  - Solves hard data association problem
- Articulated tracking:
  - Only on individual "tracklets" between occlusions
  - GP regression on full-body pose

Topics of This Lecture

- Pictorial Structures
  - Model components
  - Prior
  - Likelihood Model
- Recap: Inference
  - Sum-Product algorithm
  - Max-Sum algorithm
- Efficient Inference in Pictorial Structures
  - Generalized Distance Transform
  - Effect on Computation
- Results
Today: Pictorial Structures

- Pose estimation as inference in a graphical model
  - [Fischler & Elschlager, 1973; Felzenszwalb & Huttenlocher, 00]

Pictorial Structures

- Each body part one variable node
  - Torso, head, etc. (11 total)
- Each variable represented as tuple
  - E.g. $y_{torso} = (x, y, \mu, s)$ with
    - $x$ rotation of the part
    - $s$ scale
- Discretize label space $y$ into $L$ states
  - E.g., size of $L$ for $y = (x, y, \mu, s)$
    - $L = 125 \times 125 \times 8 \times 4 = 500\,000$
    - Efficient search needed to make this feasible!

Recap: Factor Graphs

- Joint probability
  - Can be expressed as product of factors: $p(x) = \frac{1}{Z} \prod f_i(x_i)$
  - Factor graphs make this explicit through separate factor nodes.
- Converting a directed polytree
  - Conversion to undirected tree creates loops due to moralization!
  - Conversion to a factor graph again results in a tree!

Two Model Components

- Prior $p(L)$
  - Models kinematic dependencies between body parts
  - Tree-structured prior (constraints b/w body parts) lead to efficient inference
  - Generalized distance transform provide additional efficiency
- Likelihood of body parts $p(E | L)$
  - Models possible appearances of body parts
  - Substantial improvements in recent years in appearance modeling and detection
- Finding body parts = Pose estimation
Human Body Pose Models - Prior $p(L)$
- E.g., [Felzenszwalb & Huttenlocher, IJCV'05]
- E.g., [Andriluka et al., IJCV'12]

Human Body Pose Models - Prior $p(L)$
- E.g., [Felzenszwalb & Huttenlocher, IJCV'05]
- E.g., [Andriluka et al., IJCV'12]

Kinematic Tree Prior
- Notation
  - (from [Andriluka et al., IJCV'12])
  - Body configuration $L = \{l_0, l_1, \ldots, l_N\}$
  - Each body part: $l_i = (x_i, y_i, \theta_i, s_i)$
- Prior
  - $p(L) = p(l_0) \prod_{(i,j) \in G} p(l_i | l_j)$
    - with $p(l_0)$ assumed uniform
    - with $p(l_i | l_j)$ modeled using a Gaussian

Pictorial Structures
- Potentials (= energies = factors)
  - Unaries for each body part (torso, head, ...)
  - Pairwise between connected body parts
- Body pose estimation
  - Find most likely part location $\Rightarrow$ Sum-product algorithm (marginals)
  - Find the best overall configuration $\Rightarrow$ Max-sum algorithm (MAP estimate)

Kinematic Tree Prior
- Gaussian assumption for $p(l_i | l_j)$
  - This may seem like a significant limitation.
  - E.g., distribution of forearm configuration given the upper arm is semi-circular, rather than Gaussian!
- Solution
  - Transform part configuration $l_i$ into coordinate system of the joint, where the distribution is captured well by a Gaussian:
    $$ T_{ji}(l_i) = \begin{bmatrix} x_i + s_i d_{ji}^x \cos \theta_i - s_i d_{ji}^y \sin \theta_i \\ y_i + s_i d_{ji}^y \sin \theta_i + s_i d_{ji}^x \cos \theta_i \\ \theta_i \\ s_i \end{bmatrix} $$
  - with $d_{ji}^x = \|d_{ji}\|_x$ position of the joint between parts $i$ and $j$, represented in the coordinate system of part $j$.
**Pictorial Structures: Model Components**

- Body is represented as flexible combination of parts
  - posterior over body poses
  \[ p(L|E) \propto p(E|L)p(L) \]

**Likelihood Model**

- Many variants have been proposed over the years...
  - [Felzenszwalb, IJCV'05]
    - Modeled using rectangular parts based on \(f_g/b_g\) probabilities
      - \(N_1\): #fg pixels inside rectangle
      - \(A_1\): size of rectangle
      - \(N_2\): #fg pixels inside border
      - \(A_2\): size of border area
      - \(t\): #pixels in image
    - Part likelihood
      \[ p(E|l) = q_1^{N_1} (1 - q_1^{A_1-N_1}) q_2^{N_2} (1 - q_2^{A_2-N_2}) |0.5^{t-A_1-A_2} | \]

- Assumption
  - Evidence (image features) for each part independent of all other parts
  \[ p(E|L) = \prod_{i=1}^{N} p(E|l_i) \]

- The assumption is clearly not correct, but
  - Allows efficient computation
  - Works rather well in practice
  - Training data for different body parts should cover “all” appearances
**Recap: Sum-Product Algorithm**

- **Objectives**
  - Efficient, exact inference algorithm for finding marginals.

- **Procedure**
  - Pick an arbitrary node as root.
  - Compute and propagate messages from the leaf nodes to the root, storing received messages at every node.
  - Compute and propagate messages from the root to the leaf nodes, storing received messages at every node.
  - Compute the product of received messages at each node for which the marginal is required, and normalize if necessary.

\[
p(x) \propto \prod_{x} \mu_{f_{j} \rightarrow x}(x)
\]

- **Computational effort**
  - Total number of messages \(= 2 \cdot \text{number of graph edges} \).

**Recap: Max-Sum Algorithm**

- **Objective**
  - An efficient algorithm for finding

\[
p(x_{\text{max}}) = \max \, p(x)
\]

- **Key ideas**
  - We are interested in the maximum value of the joint distribution \(p(x^{\text{max}})\).
  - For numerical reasons, use the logarithm.
  - Maximize the sum (of log-probabilities).
Recap: Max-Sum Algorithm

- Initialization (leaf nodes)
  \[ \mu_{n-1}(x) = 0 \quad \mu_{n-1}(x) = \ln f(x) \]

- Recursion
  - Messages
    \[ \mu_{n-1}(x) = \max_{v} \ln f(x, x_1, \ldots, x_n) + \sum_{u \in \text{neighbors}(v)} \mu_{u-n}(x_u) \]
    \[ \mu_{n-1}(x) = \sum_{v \in \text{children}(n)} \mu_{f_n}(x) \]
  - For each node, keep a record of which values of the variables gave rise to the maximum state:
    \[ \phi(x) = \arg \max_{H} \ln f(x, x_1, \ldots, x_n) + \sum_{u \in \text{neighbors}(v)} \mu_{u-n}(x_u) \]

Recap: Max-Sum Algorithm

- Termination (root node)
  - Score of maximal configuration
    \[ p_{\text{max}} = \max_x \sum_{v \in \text{children}(n)} \mu_{f_n}(x) \]
  - Value of root node variable giving rise to that maximum
    \[ x_{\text{max}} = \arg \max_x \sum_{v \in \text{children}(n)} \mu_{f_n}(x) \]
  - Back-track to get the remaining variable values
    \[ x_{\text{max}}^{\text{root}} = \phi(x_{\text{max}}) \]

Topics of This Lecture

- Pictorial Structures
  - Model components
  - Prior
  - Likelihood Model
- Recap: Inference
  - Sum-Product algorithm
  - Max-Sum algorithm
- Efficient Inference in Pictorial Structures
  - Generalized Distance Transform
  - Effect on Computation
- Results

Efficient Inference

- Best location given by MAP
  \[ \max_L p(L|E) = \max_L \sum_{y \in Y} p(y|L) p(e|y) \]
  \[ = \min_{y \in Y} \sum_{e \in E} \ln p(y|L) - \ln p(e|y) \]
  - Consider case of 2 parts
    \[ \min_{l_1, l_2} (\min x \ln p(l_1|l_2) - \ln p(e_1|l_1) - \ln p(l_1|l_2)) \]
  - Rename things
    \[ \min_{l_1, l_2} (m_1(l_1) + m_2(l_2) + d(l_1, l_2)) \]
  - Assume \( d \) to have quadratic form
    \[ d(l_1, l_2) = ||l_1 - T(l_2)||^2 \]
  - Then
    \[ \min_{l_1} (m_0(l_1) + m_1(l_1) + d(l_1, l_2)) \]
    \[ = \min_{l_1} (m_0(l_1) + \min_{l_2} (m_1(l_2) + d(l_1, l_2))) \]
    - with the second term a generalized distance transform (gDT).
    - Algorithms exist to compute gDT efficiently.
  - Thus
    \[ \min_{l_1} (m_0(l_1) + dT_{l_2}(T(l_2))) \]
  - Finding the best part configuration can be done sequentially, rather than simultaneously!

Distance Transform

- Given points \( p \in P \) on a grid (e.g., image) \( G \)
  - Distance Transform associates to each location \( x \in G \) the distance to the nearest point \( p \in P \)
    \[ DT_p(x) = \min_{p \in P} \{d(x, p)\} \]
    - or equivalent
      \[ DT_p(x) = \min_{q \in G} \{d(x, q) + 1(q)\} \]
      \[ 1(q) = \begin{cases} 0 & \text{if } q \in P \\ \infty & \text{otherwise} \end{cases} \]
  - Example
    \[ d(x, q) = |x - q| \]
    \[ DT_p(x) = \min_{q \in G} \{|x - q| + 1(q)\} \]

Efficient Transform
Generalized Distance Transform

- Replace binary function \( 1(q) \) with general function \( f(q) \)
  \[
  DT_f(x) = \min_{q \in \mathbb{Z}^2} \{ d(x, q) + f(q) \}
  \]
  - We can assign "soft membership of all grid elements to \( P \).
  - \( f(q) \) is sampled on the entire grid \( G \).

- In our case
  - \( f \) corresponds to \( m_1 \).
  - Distance corresponds to \( d(l_1, l_0) = ||l_1 - T(l_0)||^2 \)
  \[
  DT_{m_1}(T(l_0)) = \min_{l_1} \{ m_1(l_1) + d(l_1, l_0) \}
  \]

Example: Part Model of Motorbikes

- Model
  - 2 parts (use both wheels), simple translation between them given by \((x, y)\) position
  1. Part unaries (log prob) \(-m_0(l_0)\) and \(m_0(l)\)
  2. Distance transform of \( m_1(l_1) \)
  3. Simply find minimum of sum
     \[
     \min_{l_0} (m_0(l_0) + DT_{m_1}(T(l_0)))
     \]

Topics of This Lecture

- Pictorial Structures
  - Model components
  - Prior
  - Likelihood Model
- Recap: Inference
  - Sum-Product algorithm
  - Max-Sum algorithm
- Efficient Inference in Pictorial Structures
  - Generalized Distance Transform
  - Effect on Computation
- Results

Results

- Tracking and interpreting detailed body motion.

References and Further Reading

- Pictorial Structures
- Human Body Pose Estimation with Pictorial Structures