Human Behavior Understanding for Human-Robot Interaction
Humanoid Robot Control using Depth Camera

Halit Bener SUAY
Sonia CHERNOVA

WPI
General Pipeline

1. Data acquisition (e.g., from 3D sensors)
2. Feature extraction and representation construction
3. Robot learning: e.g., classification (recognition), clustering (knowledge discovery), or learning from demonstration
4. Decision making or planning
5. Action execution
General Pipeline

1. Data acquisition (e.g., from 3D sensors)
2. Feature extraction and representation construction
3. Robot learning: e.g., classification (recognition), clustering (knowledge discovery), or learning from demonstration
4. Decision making or planning
5. Action execution
Data Acquisition from Kinect

1. Color-depth images (i.e., RGBD images)
2. 3D point clouds
Data Acquisition from Kinect

3D point clouds and color-depth images are not equivalent!

• 3D point clouds contains more information than RGBD images or color-depth images
General Pipeline

1. Data acquisition (e.g., from 3D sensors)
2. **Feature extraction**
   and representation construction
3. Robot learning: classification (recognition), clustering (knowledge discovery), or learning from demonstration
4. Decision making or planning
5. Action execution
Feature Extraction

• Definition: feature extraction is the process of defining a set of features or attributes from raw data, which will most efficiently or meaningfully encode the information that is important for robot learning (e.g., classification, clustering, reinforcement learning) or other analysis.
Feature Extraction

• Definition: feature extraction is the process of defining a set of features or attributes from raw data, which will most efficiently or meaningfully encode the information that is important for robot learning (e.g., classification, clustering, reinforcement learning) or other analysis
  • eliminating redundancy in the raw data
  • reducing noise or outliers from the raw data
  • restructuring the raw data

• Goal: To construct a representation that
  • maximize pattern discrimination (effectiveness)
  • minimize the number of the features (efficiency)
Feature Extraction

• Examples: skeleton of 3D models
Feature Extraction

On-the-fly Curve-skeleton Computation for 3D Shapes
Andrei Sharf Thomas Lewiner Ariel Shamir Leif Kobbelt
Feature Extraction

• Examples: surface norm from 3D point clouds
Feature Extraction

Real-time Normal Estimation

Nico Blodow
Intelligent Autonomous Systems Group
Technische Universität München
Feature Extraction

• Examples: local features from interest points
Feature Extraction

• Examples: local features from interest points
Feature Extraction

Target Tracking From Aerial Images
Feature Extraction

• Example: skeleton features (directly provided by the Kinect sensor as we previously discussed)

Kinect V1

• Each joint has 3 values (joint world coordinates in meters)
• Totally 60 elements in a feature vector from each frame
• Considering orientations and other elements, we have a large feature with a lot of noise or irrelevant information
General Pipeline

1. Data acquisition (e.g., from 3D sensors)
2. Feature extraction and representation construction
3. Robot learning: e.g., classification (recognition) or clustering (knowledge discovery)
4. Decision making or planning
5. Action execution
Skeleton Data

• Skeleton information is now considered as the raw data directly provided by the Kinect sensor

• Each joint has 3 values (joint world coordinates in meters)
• Totally 60 elements in a feature vector from each frame
• Considering orientations and other elements, we have a large feature with a lot of noise or irrelevant information

How to create a high-level representation from skeleton data?
Skeleton-based representations

• Solution 1: Only use more descriptive joints
Skeleton-based representations

- Solution 2: Compute additional features
Skeleton-based representations

• Solution 2: Compute additional features
Skeleton-based representations

• Solution 2: Compute additional features
Skeleton-based representations

• How to encode spatio-temporal characteristics?
Skeleton-based representations

• How to encode spatio-temporal characteristics?

A most common method is to use HISTOGRAMS
Histogram

• A histogram is a bar chart that shows how many data points fit within a certain range
Histogram

- That range is the bin width.
- The height of a rectangle is the frequency.
Histogram

- Histogram: In statistics, a histogram is a graphical representation of the distribution of data.
Key issues in histogram

- Dealing with noise
- Dealing with different number of data instances
- Selecting hyper-parameters
- Dealing with data from multi-sources
Key issues in histogram

• Dealing with noise
• Dealing with different number of data instances
• Selecting hyper-parameters
• Dealing with data from multi-sources
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar

What is the problem you can see in this histogram?
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar

Noise?
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar

Noise?

Simply ignore the noisy data
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar

Noise?

Simply ignore the noisy data
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar

Change boundary bin range
Use as a single bin

> 200

Noise?
Histogram

• How to build a histogram?
  ▪ Show frequencies of a range of values by height of each bar

Change boundary bin range
Use as a single bin

Noise?
Key issues in histogram

• Dealing with noise
• Dealing with different number of data instances
• Selecting hyper-parameters
• Dealing with data from multi-sources
Histogram

• A histogram can be unnormalized displaying counts of data instances falling in each range/bin.

Heights of bars correspond to actual number of majors
Histogram

• A histogram can be **normalized** displaying relative frequencies.
• It then shows the proportion of data that fall into each bin.

287 students; divide by 287 to obtain relative frequency
Histogram

Heights of bars correspond to actual number

287 students; divide by 287 to obtain relative frequency
Key issues in histogram

• Dealing with noise
• Dealing with different number of data instances
• Selecting hyper-parameters
• Dealing with data from multi-sources
Histogram

What are the hyper-parameters of a histogram?
Histogram

• Choose a user-defined number of bins
Histogram

• Choose a user-defined number of bins
  • Too many bins: bins too small (range too narrow)
Histogram

• Choose a user-defined number of bins
  • Too few bins: bins too large (range too wide)
Key issues in histogram

• Dealing with noise
• Dealing with different number of data instances
• Selecting hyper-parameters
• Dealing with data from multi-sources
Histogram

• Concatenation is a most common way to integrate multiple histograms
Key issues in histogram

• Dealing with noise
• Dealing with different number of data instances
• Selecting hyper-parameters
• Dealing with data from multi-sources
Skeleton-based representations

• How to encode spatio-temporal characteristics?

A most common method is to use HISTOGRAMS
Skeleton-based representations

• Solution 1: Only use more descriptive joints
Skeleton-based representations

- Solution 2: Compute additional features
Skeleton-based representation

- Solution 2: Feature extraction - compute additional features

Representation: **Histogram of Joint Position Differences**

*Histogram of Joint Position Differences*: Given the location of a joint \((x, y, z)\) and a reference joint \((x_c, y_c, z_c)\) in the world coordinate, the joint displacement is defined as

\[
(\Delta x, \Delta y, \Delta z) = (x, y, z) - (x_c, y_c, z_c)
\]

The reference joint can be the skeleton centroid or a fixed joint. For each sequence of human skeletons representing an activity, a histogram is computed for the displacement along each dimension (e.g., \(\Delta x\), \(\Delta y\) or \(\Delta z\)). Then, the computed three histograms are concatenated into a single vector as a feature.
Skeleton-based representation

- Solution 2: Feature extraction - compute additional features

Representation: **Joint Movement Volume Features**
Skeleton-based representation

- Solution 2: Feature extraction - compute additional features

Representation: **Joint Movement Volume Features**

*Joint Movement Volume Features:* For the $j$th joint, extreme positions during the full action sequence are computed along $x, y,$ and $z$ axises. Then, maximum moving range along each dimension is computed by:

\[
L_x = \max(x_j) - \min(x_j), \\
L_y = \max(y_j) - \min(y_j), \\
L_z = \max(z_j) - \min(z_j)
\]

and joint volume is defined as:

\[
V_j = L_x L_y L_z
\]

For each joint, we incorporate the joint volume $V_j$ and $L_x, L_y, L_z$ in the feature vector.
Skeleton-based representation

- Solution 2: Feature extraction - compute additional features

Representation: **Covariance of 3D Joints**

\[
\text{Covariance of 3D Joints: Let } x_i^{(t)}, y_i^{(t)}, z_i^{(t)} \text{ be the } x, y \text{ and } z \text{ coordinates of the } i\text{th joint in frame } t. \text{ Let } S \text{ be the vector of } K \text{ joint locations:} \\
S^{(t)} = [x_1^{(t)}, \ldots, x_K^{(t)}, y_1^{(t)}, \ldots, y_K^{(t)}, z_1^{(t)}, \ldots, z_K^{(t)}] \\
\text{Then, the sample covariance of the joint positions in the trajectory } [1, T] \text{ is} \\
\text{Cov}(S) = \frac{1}{T-1} \sum_{t=1}^{T} (S^t - \bar{S})(S^t - \bar{S})^T \\
\text{where } \bar{S} \text{ is the sample mean. Since } \text{Cov}(S) \text{ is symmetric, only upper triangle values are used.}
Skeleton-based representation

• Solution 2: Feature extraction - compute additional features

Representation: **Covariance of 3D Joints**

If the entries in the column vector

\[
X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}
\]

are random variables, each with finite variance, then the covariance matrix \( \Sigma \) is the matrix whose \( (i, j) \) entry is the covariance

\[
\Sigma_{ij} = \text{cov}(X_i, X_j) = \mathbb{E}[(X_i - \mu_i)(X_j - \mu_j)]
\]

where

\[
\mu_i = \mathbb{E}(X_i)
\]

is the expected value of the \( i \)th entry in the vector \( X \). In other words,

\[
\Sigma = \begin{bmatrix}
\mathbb{E}[(X_1 - \mu_1)(X_1 - \mu_1)] & \mathbb{E}[(X_1 - \mu_1)(X_2 - \mu_2)] & \cdots & \mathbb{E}[(X_1 - \mu_1)(X_n - \mu_n)] \\
\mathbb{E}[(X_2 - \mu_2)(X_1 - \mu_1)] & \mathbb{E}[(X_2 - \mu_2)(X_2 - \mu_2)] & \cdots & \mathbb{E}[(X_2 - \mu_2)(X_n - \mu_n)] \\
\vdots & \vdots & \ddots & \vdots \\
\mathbb{E}[(X_n - \mu_n)(X_1 - \mu_1)] & \mathbb{E}[(X_n - \mu_n)(X_2 - \mu_2)] & \cdots & \mathbb{E}[(X_n - \mu_n)(X_n - \mu_n)]
\end{bmatrix}.
\]
Skeleton-based representation

• Solution 2: Feature extraction - compute additional features

Representation: **Covariance of 3D Joints**
Skeleton-based representation

- Solution 2: Feature extraction - compute additional features

Representation: **Covariance of 3D Joints**

Temporal Pyramid:
Skeleton-based representation

• Solution 2: Feature extraction - compute additional features

Representation: **Histogram of Oriented Displacement**

**Histogram of Oriented Displacements:** (1) project 3D trajectory of each joint onto the 2D Cartesian planes (i.e., $xy$, $yz$, and $zx$); (2) on each 2D Cartesian plane, compute the orientation of the trajectory to the reference coordinate; (3) compute a normalized histogram of all orientations (from all joints in adjacent time).
Details of HOD

- Project 3D skeleton data onto 3 2D-Cartesian plans.
Details of HOD

- On each plan, we consider a temporal sequence of the positions of the same joint
Details of HOD

• When we look at the temporal adjacent join positions:
Details of HOD

• We compute its orientation, i.e., the angle with respect to the x axis
Details of HOD

- Then, we can create a histogram (the original representation is based on a weighted histogram)
Details of HOD

• HOD can be considered to be speed-invariant
Details of HOD

• HOD can be considered to be scale-invariant
Details of HOD

• To better include temporal information, temporal pyramid can be applied (e.g., describing the sequence using all, halves, and quarters of data for 3-level pyramid)
Details of HOD

• 3-level pyramid
General Pipeline

1. Data acquisition (e.g., from 3D sensors)
2. Feature extraction and **representation construction**
3. Robot learning: e.g., classification (recognition) or clustering (knowledge discovery)
4. Decision making or planning
5. Action execution
• Question: Can we use the local features (e.g., “regions of corners”) to represent an observation or a human activity?
Bag-of-Words representation

• Question: Can we use the local features (e.g., “regions of corners”) to represent an observation or a human activity?
Bag-of-Words representation

• What is in the picture?
Bag-of-Words representation

Object  Bag of ‘words’
Bag-of-Words representation

- Bag-of-words representation has its origin in text processing
- Orderless document representation: frequencies of words from a dictionary
Bag-of-Words representation

- Bag-of-words representation has its origin in text processing
- Orderless document representation: frequencies of words from a dictionary

![Text-based visualization of documents with words highlighted]
Bag-of-Words representation

- Bag-of-words representation has its origin in text processing
- Orderless document representation: frequencies of words from a dictionary
Bag-of-Words representation

• Bag-of-words representation has its origin in text processing
• Orderless document representation: frequencies of words from a dictionary
Bag-of-Words representation

- Texture is characterized by the repetition of basic elements
Bag-of-Words representation

- Texture is characterized by the repetition of basic elements
Bag-of-Words representation

• Procedures:
  • Feature detection: e.g., corners

![Left Image](image1)

![Right Image](image2)

• Now we need to match pixels in the images

• To do that, take a region around a given pixel...

• And search for it in the other image

Match found!
Bag-of-Words representation

- Procedures:
  - Feature detection: e.g., corners
  - Feature description: e.g., incorporate the information of local regions around the corners
Bag-of-Words representation

• Procedures:
  • Feature detection: e.g., corners
  • Feature description: e.g., incorporate the information of local regions around the corners
  • Dictionary learning: e.g., through clustering then assign each feature descriptor an index
Bag-of-Words representation

- Procedures:
  - Feature detection: e.g., corners
  - Feature description: e.g., incorporate the information of local regions around the corners
  - Dictionary learning: e.g., through clustering then assign each feature descriptor an index
  - Bag-of-words representation
Bag-of-Words representation

• Advantages:
  • Robust to partial occlusion, illumination changes, etc.
  • Independent of global skeleton data
  • Capable of modeling more complicated scenarios, such as human-object interaction
  • Capable of capturing context information contained in the scene

How about modeling human activities?
Bag-of-Words representation

• Limitations (of the BoW discussed so far):
  • Incapable of modeling time
  • Incapable of dealing with 3D perceptual data

  • Ignore spatial information, since assuming there’s no order within the visual features/words.
Bag-of-Words representation

• Local spatio-temporal (LST) features to remove some limitations
  • In Capable of modeling time
  • In Capable of dealing with 3D perceptual data

• Ignore spatial information, since assuming there’s no order within the visual features/words.
Ethical issues in human-robot interaction