6. Introduction to digital media retrieval
The workflow of digital media analysis and retrieval

1. Digital media
2. Data stream
3. Find features
4. Recognition, classification/clustering
5. Indexing and retrieval
6. Data segmentation
3. Video retrieval techniques
Differences and relations between image and video

- Images are **static**, but video are **dynamic**.
- Video stream can be viewed as sequence of image frames.
CBVR

• Sample YouTube Video page:
Main methods of digital media retrieval

• **Text-based** digital media retrieval

• **Content-based** digital media retrieval
Why we need video shots?

a. **Text Retrieval**: Keyword Extraction

- Diagram showing the process of indexing and document storage related to text retrieval.
Why we need video shots?

b. **Database Query**: Entity Extraction

<table>
<thead>
<tr>
<th>sid</th>
<th>name</th>
<th>login</th>
<th>age</th>
<th>gpa</th>
</tr>
</thead>
<tbody>
<tr>
<td>53666</td>
<td>Jones</td>
<td>jones@cs</td>
<td>18</td>
<td>3.4</td>
</tr>
<tr>
<td>53688</td>
<td>Smith</td>
<td>smith@eecs</td>
<td>18</td>
<td>3.2</td>
</tr>
<tr>
<td>53650</td>
<td>Smith</td>
<td>smith@math</td>
<td>19</td>
<td>3.8</td>
</tr>
</tbody>
</table>
Why we need video shots?

Video Shots in Storage

Indexing

Shot Indexing

???
Video shot == keyword in video?

Shot is used as basic unit for video indexing!

Storage Database

Query Processing Server
Overview

• CBVR has two phases:
  – Database Population phase
    • Video shot boundary detection
    • Key Frames selection
    • Feature extraction
  – Video Retrieval phase
    • Similarity measure
Overview (cont.)

[Wang, Li, Wiederhold, 2001]
Structuralizing video data

- **semantic content layers**, e.g., scenes and shots in a video program.
  - These layers are erased when they are displayed for audience, which weakens the ability for user dealing with raw video data.
Fundamental definitions in video structurization

• Frame (帧)
• Shot (镜头)
• Key frame (关键帧)
• Scene (场景)
• Group (组)
Fundamental definitions in video structurization

- Frame (帧)
- Shot (镜头)
- Key frame (关键帧)
- Scene (场景)
- Group (组)
a. Scene Cuts:

Sudden change of video content or focus
Proposal

- Analyze a video stream
- Segment the stream into shots
- Index shots using extracted features
  - Camera work characteristics
  - Color representations
- Browsing methods and user interfaces
Desired Video Interaction

• Focus on fast visual browsing.

• Ability to grasp idea of lengthy video in short time.

• Not simply fast forward.

• Challenge: find and manage essential visual cues, then present them visually in an effective way.
Viewer-Video Interaction: Conceptual Model

a) Viewer Interaction

b) Video Computing

c) Video Production & Editing
Video Production

- **Key Concepts:**
  - **Take:** continuous video
  - **Cut:** separates takes
  - **Camera characteristics**
    - Pan, tilt, zoom, etc.
  - **Shot:** edited takes

- **Resulting video contains embedded info:** cut points, camera characteristics
Main Function: Make the implied video structure explicit.
Video Segmentation: Problems

- Traditional Cut Detection – detect differences between frames using inter-frame comparisons (intensity, RGB, motion vectors).
- Misdetection due to rapid object motion, slow motion, animation, strobes, fading, wiping, dissolving, etc.
- Result: Low successful detection rate.
Basic video segmentation metrics

- Pair-wise comparison
  - Pixel-level
    - Sensitive to camera movement and motion
  - Block-level (Likelihood ratio)
    - Can tolerate small motion

\[ DP_i(k, l) = \begin{cases} 
  1 & \text{if } |P_i(k, l) - P_{i+1}(k, l)| > t \\
  0 & \text{otherwise} 
\end{cases} \]

\[
\frac{\sum_{k,l=1}^{M,N} DP_i(k, l)}{M \times N} * 100 > T
\]

\[
\frac{\left[ \frac{S_i + S_{i+1}}{2} + \frac{(m_i - m_{i+1})^2}{2} \right]^2}{S_i \times S_{i+1}} > t
\]

mi: mean intensity
Si: corresponding variance
Basic video segmentation metrics

How to measure statistical property of video frames?

Color Histogram
Basic video segmentation metrics

- Histogram comparison
  - Basic
    - Tolerate motion better
  - \( \chi^2 \)-test
- Color level can also be used but only the MSB to save the number of bins

\[
SD_i = \sum_{j=1}^{G} |H_i(j) - H_{i+1}(j)|
\]

\[
SD_i = \sum_{j=1}^{G} \frac{|H_i(j) - H_{i+1}(j)|^2}{H_{i+1}(j)}
\]
Sample of using histogram
Scene Cut
Gradual transition detection
Gradual transition detection

- Twin-comparison

- Use two thresholds $T_b$ and $T_s$ to accommodate both short-term and long-term transitions

- Differences of $(F_1, F_2)$, $(F_2, F_3)$, $(F_3, F_4)$ are small, but difference of $(F_1, F_4)$ is still big
**Twin-comparison**

- $F_s$ — the potential beginning frame of the transition
- $F_e$ — the ending frame of the transition

```plaintext
scan frame
if (Diff($F_i$) ≥ $T_b$)
    detect as camera break
else if ($T_b$ > Diff($F_i$) ≥ $T_s$)
    $F_s$ ← $F_i$
    $i$ ← $i$ + 1
    while (Diff($F_i$) ≥ $T_s$)
        $i$ ← $i$ + 1
    if (Diff($F_s$, $F_i$) ≥ $T_b$)
        $F_e$ ← $F_i$
```
Threshold selection (Tb, Ts)

- The distribution of frame-to-frame differences has a high and sharp peak near the small value, which is caused by noise instead of transition and assumed to follow Gaussian distribution \((\mu, \sigma)\).
- Choose \(Tb = \mu + \alpha \sigma, \alpha \in [5, 6]\)
- Choose \(Ts\) to be greater than the mean difference and on the right slope of \(M\)
- \(Ts \in [8, 10]\), constant over samples
Multi-pass approach

- Scanning all frames could be computationally hard
- Temporal skipping is more useful
  - e.g. one out of every 10 frames
  - Better for detecting gradual transition
  - May miss camera break
  - May get false detection (distance increased)
- Multi-pass approach
  - First pass, use either pair-wise or histogram with large skip factor and smaller Tb to collect the potential regions.
  - Second pass, two methods may be applied together (hybrid) to search the candidate regions while increasing the confidence.
Distinguish camera movement

- To distinguish gradual transitions from changes made by camera movements

- Basic approach—observing **optical flow** via motion vectors

---

Fig. 6a–c. Motion vector patterns resulting from camera panning and zooming. a / Camera panning direction. b Camera zoom-out. c Camera zoom-in
Distinguish camera movement

- **Panning**
  - Distribution of motion vectors has a single modal value ($\theta_m$) that corresponds to the panning direction.

- **Zooming**
  - The vertical components of top and bottom motion vectors have different signs.
  - Similarly for horizontal components of left and right motion vectors.

\[
\sum_{k}^{N} |\theta_k - \theta_m| \leq \Theta_p
\]
Yet Another Video Segmentation

\( V = \text{image difference} \)

Video frames

\[ V(1), V(2), V(3), \ldots \]

Image difference

Filter

\[ F(V(1)), F(V(2)), F(V(3)), \ldots \]
Video Segmentation: Solution

- 92-98% success rate over 4.5 hours of video (news, movies, documentaries)
- 90% success when 1/3 of all cuts were via special effects
Shot Analysis

- Shot is simply sequence of frames capturing a scene’s spatial and temporal context.
- Extract this information:
  - Camera work yields spatial situation
  - Color info yields object information
Camera Work Information Extraction

- Camera movement causes global change in objects.
- Resulting point traces = motion vectors
- Motion vectors yield camera work parameters
- Computationally complex, not robust
Camera Work Information Extraction

- Proposal based on video x-ray imaging.
- Easy calculations, robust
Camera Work Information Extraction

- Parallel to time = fixed camera
- Slant = camera pan
- Degree of slant = speed of pan
- Line spread = zoom
- No information present for track and dolly
New Video Interfaces

- VideoScope
- VideoSpaceIcon
- ViewSpaceMonitor
- PaperVideo
PaperVideo

- Photo albums and video indexing.
- Shows potential simplicity of structured video apps.
VideoScope

- Possible use as video engineering tool.
- Shows potential complexity of structured video apps.
Related Work

- **Importance of visual interface**
  - Must activate user’s visual sense
  - Must stimulate user’s ability to manipulate video

- **What can be done in video production stage?**
Notable Reference

Cut Detection

Keyframe extraction
Reference

- Key Frame Extraction

http://www.cs.ust.hk/~rossiter/mm_projects/video_key_frame/key_frame_index.html
Reference

• Key Frame Extraction

http://www.cs.ust.hk/~rossiter/mm_projects/video_key_frame/key_frame_index.html
关键帧提取技术

- 镜头边界法
  - 选取镜头中的首帧和末帧
- 颜色特征法
  - 首帧为关键帧，其后比较与前面关键帧的颜色差异
- 运动分析法
  - 分析相机的运动
- 聚类分析法
聚类分析法

• 设一个镜头 \( S = \{ f_1, f_2, \ldots, f_m \} \)
• 找关键帧 \( [F_1, F_2, \ldots, F_n] \)
• 定义帧间距离 \( d(f_i, f_j) \)

Step 0. 设定阈值，选定初始n个关键帧位置
Step 1. 按照到关键帧的最小距离重新划分
Step 2. 指定每一聚类的中心帧为新的关键帧。

如果与上次划分区别不大则停止，否则重复Step 1和Step 2。
Brain storm
Brain storm
Brain storm
4. Graphics retrieval techniques
3D Model Similarity Search

http://infovis.uni-konstanz.de/research/projects/SimSearch3D/
Elements of polygonal mesh modeling
Triangle mesh
### Winged-Edge Meshes

#### Face List
| f0 | 4 8 9 |
| f1 | 0 10 9 |
| f2 | 5 10 11 |
| f3 | 1 12 11 |
| f4 | 6 12 13 |
| f5 | 2 14 13 |
| f6 | 7 14 15 |
| f7 | 3 8 15 |
| f8 | 4 16 19 |
| f9 | 5 17 16 |
| f10 | 6 18 17 |
| f11 | 7 19 18 |
| f12 | 0 23 20 |
| f13 | 1 20 21 |
| f14 | 2 21 22 |
| f15 | 3 22 23 |

#### Edge List
| e0 | v0 v1 f1 f12 |
| e1 | v1 v2 f3 f13 |
| e2 | v2 v3 f5 f14 |
| e3 | v3 v0 f7 f15 |
| e4 | v4 v5 f8 f16 |
| e5 | v5 v6 f9 f17 |
| e6 | v6 v7 f4 f18 |
| e7 | v7 v4 f19 |
| e8 | v0 v6 f7 f10 |
| e9 | v0 v5 f10 f1 |
| e10 | v1 v2 f3 f13 |
| e11 | v1 v6 f2 f13 |
| e12 | v2 v6 f3 f4 |
| e13 | v2 v7 f5 f4 |
| e14 | v3 v7 f6 f5 |
| e15 | v3 v8 f6 f7 |
| e16 | v6 v8 f8 f9 |
| e17 | v6 v9 f9 f10 |
| e18 | v7 v8 f10 f11 |
| e19 | v4 v8 f11 f8 |
| e20 | v1 v9 f12 f13 |
| e21 | v2 v9 f13 f14 |
| e22 | v3 v9 f14 f15 |
| e23 | v0 v9 f15 f12 |

#### Vertex List
| v0 | 0,0,0 8 9 0 23 3 |
| v1 | 1,0,0 10 11 1 20 0 |
| v2 | 1,1,0 12 13 2 21 1 |
| v3 | 0,1,0 14 15 3 22 2 |
| v4 | 0,0,1 8 15 7 19 4 |
| v5 | 1,0,1 10 9 4 16 5 |
| v6 | 1,1,1 12 11 5 17 6 |
| v7 | 0,1,1 14 13 6 18 7 |
| v8 | 5,5,0 16 17 18 19 |
| v9 | 5,5,1 20 21 22 23 |

#### Winged Edge Structure
- **Back CCW Edge**: v1 to v2
- **Front CCW Edge**: v2 to v1
- **Other Outgoing Edges**: v1 to v6, v2 to v6
- **Back CW Edge**: v2 to v1
- **Front CW Edge**: v1 to v2
Main idea

3D model → feature extraction → high dimensional feature vector → insert → ε-search
NN-search
Feature vectors

- geometry based
- image based
Feature vectors

- Geometry based

Ray-based scanning after principal axes transformation

Multi-resolution spherical harmonics representation
Feature vectors

- Image based

Flat 2D silhouettes with Fourier coefficients

Depth buffer maps from 6 directions
What’s good?

Self-organizing map of a 3D database