Content-based Image and Video Retrieval

Vorlesung, SS 2009

Shot Boundary Detection & TV Genre Classification

Hazım Kemal Ekenel, ekenel@ira.uka.de
Rainer Stiefelhagen, stiefel@ira.uka.de
CV-HCI Research Group: http://isl.ira.uka.de/cvhci

18.05.2009

Computer Vision for Human-Computer Interaction
Research Group
Outline

- Shot Boundary Detection
  - Definition
  - Types of shot boundary
  - Detection methods

- TV Genre Classification
  - Features
  - Sample systems
- Shot Boundary Detection
Video
Digital Video Processing

- Hugely important technology for archiving, content analysis, the Internet etc.
- Need for tools to support automatic browsing and retrieval of large amounts of broadcast video.
- Some Jargon.....
  - A **Frame** is 1/25 (for PAL) of a second of video.
  - A **Shot** is a sequence of frames captured by a single camera in a single continuous action.
  - A **Shot Boundary** is the transition between two shots. Can be abrupt (cut) or gradual (fade, dissolve, wipe, morph).
  - A **Scene** is a logical grouping of shots into a semantic unit.
Types of Transitions

- **Identity class:** Neither of the two shots involved are modified, and no additional edit frames are added. Hard cuts.

- **Spatial class:** Some spatial transformations are applied to the two shots involved. Wipe, page turn, slide, and iris effects.

- **Chromatic class:** Some color space transformations are applied to the two shots involved. Fade and dissolve effects.

- **Spatio-Chromatic class:** Some spatial as well as some color space transformations are applied to the two shots involved. Morphing effects.
Types of Transitions

Cut

Fade Out/In

Dissolve

Wipe

Clock Wipe
Why do we need Shot Boundary Detection?

- Shots are basic units of a video. They are required for further video analysis, such as
  - Person tracking, identification,
  - High-level feature detection …

- They provide cue about high-level semantics
  - In video production each transition type is chosen carefully to support the content and context.
  - For example, dissolves occur much more often in feature films and documentaries than in news, sports and shows. The opposite is true for wipes.
Hard Cuts

- The most common transition type.
- Direct concatenation of two shots, \( S_1(x, y, t) \) & \( S_2(x, y, t) \)

\[
S(x, y, t) = (1 - u_1(t - t_{hardcut})) \cdot S_1(x, y, t) + u_1(t - t_{hardcut}) \cdot S_2(x, y, t)
\]

- Produces a temporal visual discontinuity.
- **How to measure the discontinuity?**

\( t_{hardcut} \): Time stamp of the first frame after the hard cut

\( u_1(t) \): The unit step function
Features to Measure Visual Discontinuity

- Pixel differences
- Statistical differences
- Histograms
- Compression differences
- Edge differences
- Motion vectors
Pixel differences

- **Two common approaches:**
  
  1. Calculate pixel-to-pixel difference & Compare the sum with a threshold
  
  2. Count the number of pixels that change in value more than some threshold & Compare the total number against a second threshold

- **Sensitive to camera & object motion!**
  
  - Use an average filtering
  
  - Motion compensation
Camera Motion & Object Motion
Absolute Pixel Differences with & w/o Motion Compensation

Frame 66

Frame 69

Absolute difference w/o motion compensation

Absolute difference with motion compensation
Motion Estimation

- Adjacent frames are similar and changes are due to
  - object or
  - camera motion
Optical Flow

Assumptions:
- **color constancy**: a point in “t-1” looks the same in “t”
  - For grayscale images, this is **brightness constancy**
- **small motion**: points do not move very far
Optical Flow Constraint Equation

- Assume brightness of patch remains same in both images:
  \[ I(x + u \, \delta t, y + v \, \delta t, t + \delta t) = I(x, y, t) \]

- Assume small motion (Taylor expansion of left-hand-side upto first order):
  \[ I(x, y, t) + \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} = I(x, y, t) \]
Optical Flow Constraint Equation

\[ \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} = 0 \]

Divide by \( \delta t \) and take the limit \( \delta t \to 0 \)

\[ \frac{dx}{dt} \frac{\partial I}{\partial x} + \frac{dy}{dt} \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0 \]

Constraint Equation

\[ I_x u + I_y v + I_t = 0 \]

NOTE: \((u, v)\) must lie on a straight line

We can compute \( I_x, I_y, I_t \) using gradient operators!
A sample optical flow output

Image $I$

Image $I$ - Rotated

Illustration of optical flow

Absolute difference w/o motion compensation

Absolute difference with motion compensation
Motion Estimation Methods

- **Feature/Region Matching**: Motion is estimated by correlating/matching features (e.g., edges) or regional intensities (e.g., block of pixels) from one frame to another.
  - Block Matching
  - Phase Correlation

- **Gradient-based Methods**: Motion is estimated by using spatial and temporal changes (gradients) of the image intensity distribution and the displacement vector field.
  - Lucas-Kanade
Statistical differences

- Divide image into regions
- Compute statistical measures from these regions (e.g., mean, standard deviation ...)
- Compare the obtained statistical measures
Histogram comparison

- The most common method used to detect shot boundaries.
- Provides good trade-off between accuracy and speed.
- The simplest histogram method computes gray level or color histograms of the two images. If the bin-wise difference between the two histograms is above a threshold, a shot boundary is assumed.
- Several extensions available: Using regions, region weighting, different distance metrics …
Compression differences

- Use differences in the discrete cosine transform (DCT) coefficients of JPEG compressed frames as the measure of frame similarity.

Avoid the need to decompress the frames
Edges/Contours

- The edges of the objects in the last frame before the hard cut usually cannot be found in the first frame after the hard cut,
- The edges of the objects in the first frame after the hard cut in turn cannot be found in the last frame before the hard cut.

➤ Use Edge Change Ratio (ECR) to detect hard cuts!
Edge Change Ratio (ECR)

\[ ECR_n = \max(X_{in}^n / p_n, X_{out}^{n-1} / p_{n-1}) \]

- \( p_n \): The number of edge pixels in frame \( n \)
- \( X_{in}^n \): The number of entering edge pixels in frame \( n \)
- \( X_{out}^{n-1} \): The number of exiting edge pixels in frame \( n-1 \)

- To make the measure more robust to object motion:
  - Edge pixels in one image which have edge pixels nearby in the other image (e.g. within 6 pixels’) are not regarded as entering or exiting edge pixels.
Edge Change Ratio (ECR)
Motion

- Use motion vectors to determine discontinuity.
Fade Detection

- A fade sequence $S(x,y,t)$ of duration $T$: scaling the pixel intensities/colors of a video sequence $S_1(x,y,t)$ by a temporally monotone monotone scaling function $f(t)$

$$S(x,y,t) = f(t) \cdot S_1(x,y,t), \quad t \in [0, T]$$

- **Fade in**: $f(0) = 0$ and $f(T) = 1$
- **Fade out**: $f(0) = 1$ and $f(T) = 0$
- **Often $f(t)$ is linear**
  - **Fade in**: $f(t) = t/T$,
  - **Fade out**: $f(t) = (T-t)/T$
Fade Detection – Standard deviation of pixel intensities

\[
\text{Var}(S(x,y,t)) = \text{Var}(f(t) \times S_1(x,y,t)) \\
= f^2(t) \times \text{Var}(S_1(x,y,t)) \\
= f^2(t) \times \text{Var}(S_1(x,y)) \\
\sigma(S(x,y,t)) = f(t) \times \sigma(S_1(x,y))
\]

Method:
- Detect the monochrome frames
- Search in both directions for a linear increase in the pixels’ intensity/color standard deviation
Dissolve Detection

- A dissolve sequence $D(x,y,t)$ of duration $T$: mixture of two video sequences $S_1(x,y,t)$ and $S_2(x,y,t)$, where the first sequence is fading out while the second is fading in

$$D(x,y,t) = f_1(t) \cdot S_1(x,y,t) + f_2(t) \cdot S_2(x,y,t), \quad t \in [0, T]$$

- $f_1(t) = (T - t) / T = 1 - f_2(t)$
- $f_2(t) = t / T$

Method

- Train support vector machines
Fade out/in vs. Dissolve

Fade out/in (FOI)

Frame 2139  Frame 2140  Frame 2141

Frame 2142  Frame 2143  Frame 2144

Frame 2145  Frame 2146  Frame 2147

Frame 2148  Frame 2149  Frame 2150

Frame 2176  Frame 2177  Frame 2178

Dissolve

Frame 7531  Frame 7532

Frame 7533  Frame 7534

Frame 7535  Frame 7535
Shot Boundary Detection @ TRECVID Evaluations

- A video retrieval evaluation campaign from the National Institute of Standards and Technology (NIST), US.
- Promote progress in content-based analysis, detection, retrieval in large amount of digital video
  - Combine multiple errorful sources of evidence
  - Achieve greater effectiveness, speed, and usability
- Confront systems with unfiltered data and realistic tasks
- Measure systems against human abilities
Shot Boundary Detection @ TRECVID Evaluations

- Evaluated each year from 2001 – 2007
- 57 different research groups worldwide

<table>
<thead>
<tr>
<th>Year</th>
<th>Hours</th>
<th>Files</th>
<th>Frames</th>
<th>Trans</th>
<th>%Cut</th>
<th>%Diss.</th>
<th>%Fade</th>
<th>%Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001</td>
<td>5.8</td>
<td>42</td>
<td>594,170</td>
<td>3176</td>
<td>65.0</td>
<td>30.7</td>
<td>1.7</td>
<td>2.6</td>
</tr>
<tr>
<td>2002</td>
<td>4.8</td>
<td>18</td>
<td>545,068</td>
<td>2090</td>
<td>70.1</td>
<td>24.4</td>
<td>3.0</td>
<td>2.4</td>
</tr>
<tr>
<td>2003</td>
<td>6</td>
<td>13</td>
<td>596,054</td>
<td>3734</td>
<td>70.7</td>
<td>20.2</td>
<td>3.1</td>
<td>5.9</td>
</tr>
<tr>
<td>2004</td>
<td>6</td>
<td>12</td>
<td>618,409</td>
<td>4806</td>
<td>57.7</td>
<td>31.7</td>
<td>4.8</td>
<td>5.7</td>
</tr>
<tr>
<td>2005</td>
<td>7</td>
<td>12</td>
<td>744,604</td>
<td>4535</td>
<td>60.8</td>
<td>30.5</td>
<td>1.8</td>
<td>6.9</td>
</tr>
<tr>
<td>2006</td>
<td>7.5</td>
<td>13</td>
<td>597,043</td>
<td>3765</td>
<td>48.7</td>
<td>39.9</td>
<td>1.3</td>
<td>10.1</td>
</tr>
<tr>
<td>2007</td>
<td>6</td>
<td>17</td>
<td>637,805</td>
<td>2317</td>
<td>90.8</td>
<td>5.4</td>
<td>1.0</td>
<td>3.7</td>
</tr>
</tbody>
</table>
Cut vs. Gradual Transition Performance

Cuts

Gradual Transitions

Content-based image and video retrieval
A short break 😊
TV Genre Classification

- Multimedia content annotation
- Key issue in current convergence of audiovisual entertainment and information media
- Good information and communication technologies available
- *but* multimedia classification not mature enough
- Lack of good automatic algorithms
- Main challenge: combine and map low-level descriptors and high-level concepts
Sample Genres
Subgenres
Sample Feature - Scene Length

News Cast

Sports - Tennis

Commercials

Cartoon
Sample Feature - Audio Statistics: Wave Forms

News Cast

Sports - Race

Sports - Tennis

Commercials

Cartoon
Sample Feature - Audio Statistics: Frequency Spectrum

News Cast

Sports - Race

Sports - Tennis

Commercials
A sample system

TV Genre Classification Using Multimodal Information and Multilayer Perceptrons

Credit: Tomas Semela

Feature Sets

- Modality information in broadcast domain concerns
  - Physical properties perceived by users like colours, shapes and motion
  - Structural-syntactic information, e.g. relationships between frames, shots and scenes
  - Cognitive information related to high-level semantic concepts like faces
  - Aural analysis of noise and speech

resulting in a feature vector \( PV = (V_c, S, C, A) \)
Feature Sets

- Low-level visual feature vector component
  - Color represented by
    - hue (H)
    - saturation (S)
    - value (V)
  - Luminance (Y) represented by a grey scale [16, 233]
  - Textures described through contrast (C) and directionality (D) Tamura’s features
  - Temporal activity information (T) based of displaced frame difference (DFD)
  - 65- bin histogram for each feature
  - Last bin collects undefined values
Feature Sets

- Computed on a frame by frame basis
- Accumulated over the number of frames
- Each histogram modeled by a 10-component Gaussian mixture model
- Each component being a Gaussian distribution with three parameters
  - weight, mean and standard deviation

\[
\phi_{\mu_i, \sigma_i^2}(x) = \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}}
\]
Feature Sets

- Gaussian mixture model
  \[ \sum_{i=1}^{10} w_i \phi_{\mu_i, \sigma_i^2} \]

- example of 4 component gaussian mixture
- different means and standard deviation

resulting into a 210 – dimensional feature vector

\[ V_c = (H, S, V, Y, C, D, T) \]
Structural feature vector component

- Extracted using a shot detection module
- S1 captures information about the rhythm of the video:

\[
S_1 = \frac{1}{F_r N_s} \sum_{i=1}^{N_s} \Delta s_i
\]

- \(F_r\) is the frame rate (i.e. 25 fps), \(N_s\) total number of shots
- \(\Delta s_i\) shot length, measured as the number of frames within the \(i^{th}\) shot.
Structural feature vector component

- S2 describes shot lengths distributed along the video
  - represented by a 65-bin histogram
  - 64 bins for shot lengths [0,30s]
  - 65th bin for shots longer than 30s
  - histogram normalized by $N_s$ so the area sums to one

resulting into a 66-dimensional feature vector

$$S = (S_1, S_2)$$
Cognitive feature vector component

- Built by applying face detection
- Leads to three features

\[ C_1 = \frac{N_f}{D_P} \]

\( N_f \) total number of faces

\( D_P \) total number of frames

- \((C_2)\) describes how faces are distributed along the video
  - expressed by a 11-bin histogram
  - \( i^{th} \) bin contains the number of frames with \( i \) faces,
  - \( 11^{th} \) bin contains the number of frames with 10 or more faces
Cognitive feature vector component

- $(C_3)$ describes how faces are positioned along the video
  - 9-bin histogram where the $i^{th}$ bin represents the $i^{th}$ positions in the frame
  - Positions are top-left, top-right, bottom-left, bottom-right, left, right, top, bottom and center
  - All histograms normalized by $N_f$ so their area sums to one

resulting into a 21-dimensional feature vector

$$C = (C_1, C_2, C_3)$$
Aural feature vector component

- Derived by audio analysis of the TV programme
- Audio signal segmented into seven classes: *speech, silence, noise, music, pure speaker, speaker plus noise, speaker plus music*
- $A_1$ duration values, normalized by total duration of the video for the seven classes
- $A_2$ the average speech rate, computed from speech content transcriptions using a speech-to-text engine

resulting into a 8-dimensional feature vector

$$A = (A_1, A_2)$$
Genre Classification

- $p$ is the TV programme to be classified
- $\Omega = \{ \omega_1, \omega_2, \ldots, \omega_{N_\omega} \}$ the set of available genres
- Feature vector of $p$ is derived like described in previous slides
- Each feature vector of $p$ is input of an Neural Network
Genre Classification

- Each Neural Network has an output vector
  \[ \Phi^{(p,n)} = \{ \phi_1^{(p,n)}, \ldots, \phi_{N_\omega}^{(p,n)} \}, \ n = 1, \ldots, 4 \]
- \( \phi_i^{(p,n)} \) can be interpreted as the membership value of \( p \) to genre \( i \), according to the pattern vector part \( n \)
- Outputs combined into a resulting vector
  \[ \Phi^{(p)} = \{ \phi_1^{(p)}, \ldots, \phi_{N_\omega}^{(p)} \} \]
  where:
  \[ \Phi_i^{(p)} = \frac{1}{4} \sum_{n=1}^{4} \phi_i^{(p,n)} \]
- The genre \( j \) is selected corresponding to the maximum element of \( \phi^{(p)} \)
Experimental Results - Dataset

- About 110 hours of complete TV programs
- Genres: cartoons, football, talk show, weather forecast, news, music videos, commercials
- Each TV program manually annotated
- Dataset split into $K = 6$ disjoint subsets of equal size
- K-fold cross validation is used
Sample Clips from the Data Set

Cartoon

Commercial

Football

Music

News

Talk show

Weather forecast
Experimental Results - Settings

- All networks with one hidden layer, seven output neurons with sigmoid activation functions in the range of [0,1]
- All hidden neurons have symmetric sigmoid activation functions in the range of [-1,1]
- Aural network has 8 input neurons and 32 hidden neurons
- Cognitive network has 21 input neurons and 32 hidden neurons
- Structural network has 65 input neurons and 8 hidden neurons
- Visual network has 210 input neurons and 16 hidden neurons
Experimental Results

- Obtained accuracy with an averaged value of 92%
- In some cases even greater than 95%
- Some news - talk shows and commercials - music clips confused with each other
- Music genre shows the most scattered results due to structural, visual and cognitive inhomogenity

<table>
<thead>
<tr>
<th>Genre</th>
<th>Talk Shows</th>
<th>Commercials</th>
<th>Music</th>
<th>Cartoons</th>
<th>Football</th>
<th>News</th>
<th>Weath.For.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Talk Shows</td>
<td>91.7</td>
<td>0</td>
<td>0</td>
<td>1.6</td>
<td>0</td>
<td>6.7</td>
<td>0</td>
</tr>
<tr>
<td>Commercials</td>
<td>1.5</td>
<td>91</td>
<td>4.5</td>
<td>0</td>
<td>0</td>
<td>1.5</td>
<td>1.5</td>
</tr>
<tr>
<td>Music</td>
<td>1.7</td>
<td>5</td>
<td>86.7</td>
<td>5</td>
<td>1.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Cartoons</td>
<td>0</td>
<td>0</td>
<td>3.4</td>
<td>94.9</td>
<td>0</td>
<td>0</td>
<td>1.7</td>
</tr>
<tr>
<td>Football</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>News</td>
<td>11.1</td>
<td>0</td>
<td>0</td>
<td>1.6</td>
<td>0</td>
<td>87.3</td>
<td>0</td>
</tr>
<tr>
<td>Weath.For.</td>
<td>1.5</td>
<td>1.5</td>
<td>0</td>
<td>1.6</td>
<td>0</td>
<td>0</td>
<td>95.4</td>
</tr>
</tbody>
</table>
### Experimental Results – Comparison

<table>
<thead>
<tr>
<th>Authors</th>
<th>Clip Duration [seconds]</th>
<th>Genre Duration [minutes]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dinh et al. [4]</td>
<td>1</td>
<td>news (26), concerts (15), cartoons (19), commercials (18), music shows (11), motor racing games (21).</td>
</tr>
<tr>
<td>Xu et al. [29]</td>
<td>300</td>
<td>news (60), commercials (60), sports (60), music (60), cartoons (60).</td>
</tr>
<tr>
<td>Liu et al. [13]</td>
<td>1.5</td>
<td>news (20), commercials (20), football (20), basketball (20), weather forecasts (20).</td>
</tr>
<tr>
<td>Truong et al. [28]</td>
<td>60</td>
<td>news (96), music (96), sports (96), commercials (96), cartoons (96).</td>
</tr>
<tr>
<td>Dimitrova et al. [3]</td>
<td>60</td>
<td>training (26), testing (35).</td>
</tr>
<tr>
<td>Roach et al. [19]</td>
<td>30</td>
<td>news (1), cartoons (7), sports (7).</td>
</tr>
<tr>
<td>This work</td>
<td>120 to 2640</td>
<td>music (233), commercials (209), cartoons (1126), football (1053), news (1301), talk shows (2650), weather forecasts (120).</td>
</tr>
</tbody>
</table>
## Experimental Results - Comparison

<table>
<thead>
<tr>
<th>Authors</th>
<th>Recognised genres</th>
<th>Classifier type</th>
<th>Dataset Size [minutes]</th>
<th>Data Validation Strategy</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dinh et al. [4]</td>
<td>6</td>
<td>k-NN</td>
<td>110</td>
<td>HOV</td>
<td>96%</td>
</tr>
<tr>
<td>Dinh et al. [4]</td>
<td>6</td>
<td>DT</td>
<td>110</td>
<td>HOV</td>
<td>91%</td>
</tr>
<tr>
<td>Dinh et al. [4]</td>
<td>6</td>
<td>SVM</td>
<td>110</td>
<td>HOV</td>
<td>90%</td>
</tr>
<tr>
<td>Xu et al. [29]</td>
<td>5</td>
<td>GMM</td>
<td>300</td>
<td>HOV</td>
<td>86%</td>
</tr>
<tr>
<td>Liu et al. [13]</td>
<td>5</td>
<td>HMM</td>
<td>100</td>
<td>HOV</td>
<td>85%</td>
</tr>
<tr>
<td>Truong et al. [28]</td>
<td>5</td>
<td>DT</td>
<td>480</td>
<td>HOV</td>
<td>83%</td>
</tr>
<tr>
<td>Liu et al. [12]</td>
<td>5</td>
<td>ANN</td>
<td>100</td>
<td>HOV</td>
<td>71%</td>
</tr>
<tr>
<td>Dimitrova et al. [3]</td>
<td>4</td>
<td>HMM</td>
<td>61</td>
<td>HOV</td>
<td>85%</td>
</tr>
<tr>
<td>Roach et al. [19]</td>
<td>3</td>
<td>GMM</td>
<td>15</td>
<td>LOOCV</td>
<td>94%</td>
</tr>
<tr>
<td><strong>This work</strong></td>
<td><strong>7</strong></td>
<td><strong>MLPs</strong></td>
<td><strong>6692</strong></td>
<td><strong>KFCV</strong></td>
<td><strong>92%</strong></td>
</tr>
</tbody>
</table>
References


Questions?