Chapter 2
Image Similarity

Distributed Algorithms
for Image and Video Processing

Content

- Image Similarity
  - pixel based algorithms
  - histograms
  - aggregated image data
  - edge based algorithms (Canny)
- Application: Cut detection in videos
  - definition: cut
  - classification of a cut
  - automatic cut detection
  - quality measurements
  - detection results

Image Similarities (I)

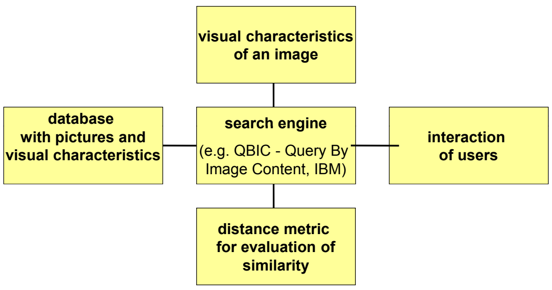


Image Similarities (II)

Central component: distance metric for the calculation of two images' similarity

- pixel based algorithms
- histograms
- aggregated image data
- edge based algorithms

Pixel Based Algorithms

Sum of absolute pixel differences

$$D_{SAD} = \frac{1}{N_x \cdot N_y} \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} |I_i(x,y) - I_j(x,y)|.$$

- images: I_i, I_j
- pixel values of the image I_i at position (x,y): $I_i\,(x,y)$
- image width, image height: N_x, N_y

Histograms (I)

Idea

- For each gray, respectively color, value, a histogram counts the absolute or relative number of pixels for this particular lightness, respectively color, in the image.

Size of a histogram (number of bins)

- 8-bit gray scale images: 256 elements
- Color pictures with 24-bit color intensity: > 16 million elements

Solution

- single histograms for each color channel

  → Decrease the number of different colors in the histogram (removal of the least significant bits for each color channel)

Histograms (II)

Histogram difference

- Minkowski-metric:

$$L_p(H_1, H_2) = \left(\sum_{m=1}^{M} |H_1(m) - H_2(m)|^p \right)^{\frac{1}{p}}$$

- *P=1*: L1-norm: sum of absolute histogram differences
- *P=2*: L2-norm (Euclidian norm): sum of squared histogram differences

Aggregated Image Data (I)

Standard deviation of the brightness value of all pixels of an image *I*

$$\sigma_I = \sqrt{\frac{1}{N_x \cdot N_y} \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (I(x,y) - \bar{I})^2}.$$

- Suitable for brightness variations

Aggregated Image Data (II)

Edge contrast

- weak edges
$$w_I = \sum_{x,y} \begin{cases} I(x,y) & \text{if} \quad \theta_w \le I(x,y) < \theta_s, \\ 0 & \text{else} \end{cases}$$
- sharp edges
$$s_I = \sum_{x,y} \begin{cases} I(x,y) & \text{if} \quad I(x,y) \ge \theta_s, \\ 0 & \text{else} \end{cases}$$

$$EC = 1 + \frac{s_I - w_I - 1}{s_I + w_I + 1}, \qquad EC \in [0, 2].$$

Aggregated Image Data (III)

Edge contrast

$$EC = 1 + \frac{s_I - w_I - 1}{s_I + w_I + 1}, \qquad EC \in [0, 2].$$

Percentage of sharp and weak edges	EC
$s_I = 0$	0
$s_I < w_I$	$0 < EC < 1$
$s_I \approx w_I > 0$	1
$s_I > w_I$	$1 < EC < 2$
$s_I \gg w_I$	2

Edge based algorithms: Canny (I)

Idea of edge based algorithms

- Analysis of the differences of two images' edges

Calculation of an edge image: Canny edge detector

- Edges emerge from brightness variations between adjacent pixels

1) In order to reduce image noise, the image is smoothed with an approximation of the Gaussian distribution:

Filter mask: $\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$

Edge based algorithms: Canny (II)

2. The single pixels' gradients are determined with the *Sobel* operator:

Mask: $\begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$ GradX $\quad \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$ GradY

3. Edge strength (gradient magnitude):
GradMag = sqrt (GradX * GradX + GradY + GradY)

Edge based algorithms: Canny (III)

- Calculate edge strength for each pixel:

$$\text{gradient orientation} = \begin{cases} \arctan\left(\frac{Grad_Y}{Grad_X}\right) & \text{if} \quad Grad_X \neq 0 \\ 0^0 & \text{if} \quad Grad_X = 0, Grad_Y = 0 \\ 90^0 & \text{if} \quad Grad_X = 0, Grad_Y \neq 0 \end{cases}$$

4 edge directions are possible:

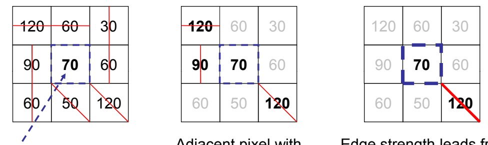
0˚, 45˚, 90˚, and 135˚

Edge based algorithms: Canny (IV)

- Edges should have a width of 1 pixel:
  → Search for local maxima (non maxima suppression)
- Edge pixel is preserved if:
  – Current edge pixel has higher edge strength than all adjacent pixels **or**
  – there is at least on adjacent pixel with a higher edge strength than the current pixel for which:
    the edge strength leads from the adjacent pixel with the higher edge strength to the current pixel.

Edge based algorithms: Canny (V)

Example: Search for local maxima



edge strength of the current pixel

Edge strength

Adjacent pixel with higher edge strength

Edge strength leads from the adjacent pixel to the current pixel
→ edge pixel is kept

Edge based algorithms: Canny (VI)

Determination of edge pixels (hysteresis)

- 2 threshold values are defined: T_{low} and T_{high}
- in the case of edge strength > T_{high}:
  → apply pixel as edge pixel
- in the case of T_{low} < edge strength < T_{high} and pixel adjoins to an edge pixel:
  → apply pixel as edge pixel

Result: edge image

Edge based algorithms: Canny (VII)

Example: Canny-edge detector

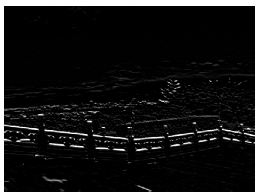
original image

smoothed image

1. Smooth image

Edge based algorithms: Canny (VIII)

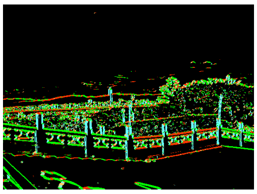
gradient in x-direction



gradient in y-direction

2. Images of gradients

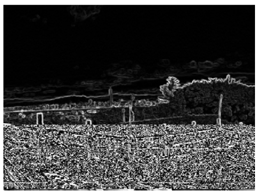
Edge based algorithms: Canny (IX)



0 degree, 45 degree, 90 degree, 135 degree

3. Edge strength   4. Edge direction

Edge based algorithms: Canny (X)



without search for local maxima   search for local maxima

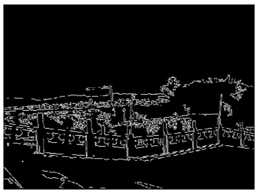
5. Search for local maxima

Edge based algorithms: Canny (XI)

edge image without threshold values   weak and sharp edges

6a. Edge image before hysteresis

Edge based algorithms: Canny (XII)



selected weak and sharp edges   edge image

6b. Edge image after hysteresis

Application: Cut detection in videos

- Definition: cut
- Classification of a cut
- Algorithm for automatic cut detection
- Evaluation of cut detection algorithms
- Results for the detection of abrupt and gradual transitions

Definition: cut

Cut
- delivers information about the production process of a movie
- separates continuous exposures which are referred to as shots
- is a movie's smallest unit in which the temporal dimension is contained
- is condition for the procedure of a movie's analysis

Scene
- content-wise similar and temporally sequenced camera shot

Dialogue
- scene in which the image switches repeatedly between two or several persons

Classification of a cut (I)

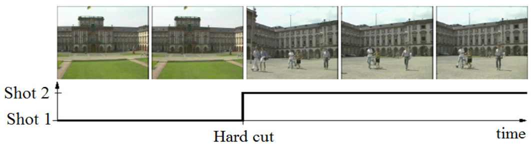
Abrupt transition
- No transition between two cuts

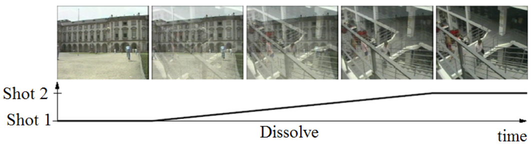
Gradual transition
- Artificial transition between two cuts
- *Dissolve*
  Continual transition between two cuts
- *Flash* and *fade*
  Dissolve in which one of the two cuts consists of monochrome – commonly black - images
- *Wiping* effect
  Pixel of selected regions are changed directly

Classification of a cut (II)

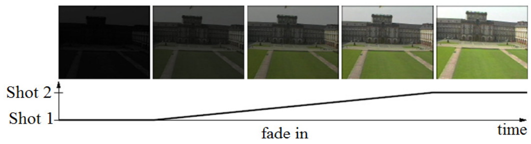
	Duration of the cut	
	Cut between two adjacent frames	Cut within a time interval
Direct change of (selected) pixels between adjacent frames	hard cut	wipe
Continuous change of all pixels between several frames		dissolve fade in /fade out

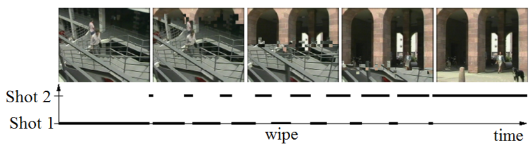
Classification of a cut (III)





Classification of a cut (IV)





Automatic cut detection (I)

Idea
- Consider similarity between two adjacent images in the video
- In general, images of one cut have a considerably higher similarity than images of different cuts

Approach
- Calculate difference between two adjacent images in each case
- Smooth the results in order to reduce classification errors through noise
- High differences indicate an abrupt transition

But
- Detection of gradual transitions hardly possible because only slight differences occur between adjacent images

Automatic cut detection (II)

Detection of abrupt transitions
- Analyze similarity of two images I_i and I_j with $1 \leq i < j \leq N$ within a video ($I_1 \ldots I_N$)

Algorithm
1. Define mapping τ , which transforms an image I_i into a feature space:
$$\tau : \mathbb{N}^m \longrightarrow F,$$
$\mathbb{N}^m$ space which is stretched through all images.
F specifies a feature space with
$$\tau(I_n) \in F. \qquad (I_n \in \mathbb{N}^m)$$

Automatic cut detection (III)

Detection of abrupt transitions

2. Define robust distance measure *D* for the estimation of the similarity of two images based on the characteristic values:

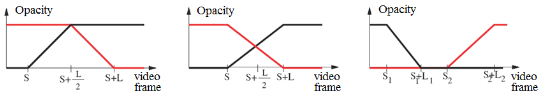
$$D : F \times F \rightarrow \mathbb{R}^+.$$

3. An abrupt transition exists between the images *i* and *j* in case that the distance of two images exceeds a threshold value *T* :

$$D_{i,j} \;=\; D\left(\tau(I_i), \tau(I_j)\right) \;>\; T_{i,j}.$$

Automatic cut detection (IV)

Detection of gradual transitions



Opacity = 1 - transparency

additive dissolve **cross dissolve** **fade in / fade out**

S: start time of soft cut, L: duration of soft cut

Assumptions

- Change of transparency can be approximated by linear function
- No changes of image content in both shots

Automatic cut detection (V)

Detection of gradual transitions

- Image content during a dissolve:

$$I_k = \alpha_k \cdot I_S + \beta_k \cdot I_{S+L} \quad \text{with} \quad S \leq k \leq S + L.$$

- During additive dissolve:

$$\alpha_k = \begin{cases} 1 & \text{for } S \leq k \leq S + L/2 \\ 1 - \frac{k-S-L/2}{L/2} & \text{for } S + L/2 < k \leq S + L \end{cases}$$

$$\beta_k = \begin{cases} \frac{k-S}{L/2} & \text{for } S \leq k \leq S + L/2 \\ 1 & \text{for } S + L/2 < k \leq S + L, \end{cases}$$

Automatic cut detection (VI)

Detection of gradual transitions

- During cross-dissolves:
  $\alpha_k = 1 - \frac{k-S}{L}$ and $\beta_k = 1 - \alpha_k$ with $S \leq k \leq S + L.$
- Fades are modeled like additive dissolves. Instead of black images $\beta_k = 0$ can also be set.

Rules for the detection of gradual transitions

- Equal modification of the distance measures between two adjacent images respectively:
  $D_{i,i+1} \approx D_{i+1,i+2} \quad \forall \quad S \leq i < S + \frac{L}{2} - 1$ and $S + \frac{L}{2} \leq i < S + L - 1.$
- Difference increases with rising distance between the images:
  $D_{i,i+j} < D_{i,i+k} \quad \forall \quad S \leq i < i + j < i + k \leq S + L.$

Pixel-based techniques (I)

Sum of absolute pixel differences

$$D_{SAD} = \frac{1}{N_x \cdot N_y} \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} |I_i(x,y) - I_j(x,y)|.$$

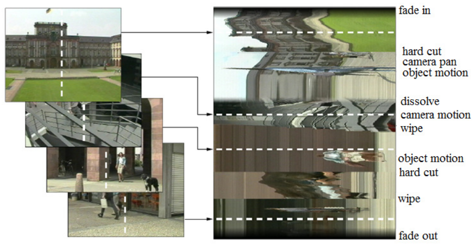
- Abrupt transition if: $D_{SAD} > T$

Advantages / Disadvantages

- Generally, very robust and reliable results. However, high error rates concerning intensive object- and camera movements.
- Slight complexity and therefore quickly calculable.

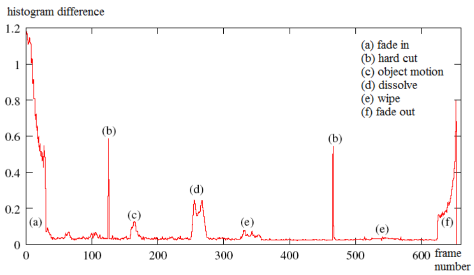
Pixel-based techniques(II)

Alternative: spatial temporal images



selected frames spatial temporal image

Histograms (I)



Histograms (II)

Advantages / Disadvantages

- Robust and reliable results
- High error rates especially in case of brightness changes or quick motion of large objects
- Low complexity of the algorithm

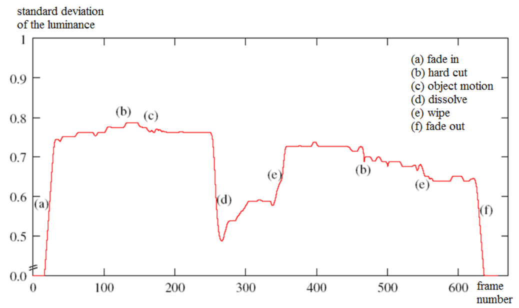
Aggregated Image Data (I)

Standard deviation of the brightness values of all pixels of an image

$$\sigma_I = \sqrt{\frac{1}{N_x \cdot N_y}\sum_{x=1}^{N_x}\sum_{y=1}^{N_y}(I(x,y)-\bar{I})^2}.$$

- The standard deviation decreases in case of fades and in the center of a dissolve because the pixels in these areas take on average brightness respectively color values.
  → Detection of fades and dissolves

Aggregated Image Data (II)



Aggregated Image Data (III)

Problems

- Fast camera and object movements:
  - Shots are blurred frequently
  - Standard deviation decreases
  - Many erroneously recognized cuts
- Reasons for blurring
  - caused through the recording (change of focus)
  - through the compression of the digital video

Aggregated Image Data (IV)

Edge contrast

$$EC = 1 + \frac{s_I - w_I - 1}{s_I + w_I + 1}, \quad EC \in [0,2].$$

Advantages / Disadvantages

- Slight influence of the camera respectively object movement
- Low complexity of the calculation
- Higher error rates occur during quick movements, because thereby the image often loses sharpness

Edge-based techniques: ECR (I)

Edge change ratio (edge change ratio, edge change fraction)

- Edges of two images *i* and *j* of a video are ascertained with the aid of the Canny edge detector
- Edge pixels, which are contained in the first but not in the second image, are indicated as *incoming* edge pixels, the edge pixels coming along in the second image as *outgoing* edge pixels.

$$\rho_{out}(i) = \frac{E_{out}(i)}{S_i}$$
$$\rho_{in}(j) = \frac{E_{in}(j)}{S_j}$$
$$ECR_{i,j} = \max\{\rho_{out}(i), \rho_{in}(j)\}.$$

- Number of incoming and outgoing edge pixels: $E_{out}(i)$, $E_{in}(i)$
- Amount of incoming and outgoing edge pixels ($\rho_{out}(i)$, $\rho_{in}(i)$) to the total amount of edge pixels S_i

Edge-based techniques: ECR (II)

Problem

- Camera and object movements elevate the number of incoming and outgoing edges significantly
  → compensation of the camera movement (see next chapter)
- Slight shift of two images' edges possible
  → edges of the image are expanded (dilatation)

Dilatation / Erosion

- Operators which are defined for a grayscale picture / and a structural element *B* with radius *r* (e.g., circle or square) as follows:

$$\text{Dilatation:} \quad (I \circ B)(x) = \sup\{I(x-r), r \in B\}$$
$$\text{Erosion:} \quad (I \bullet B)(x) = \inf\{I(x-r), r \in B\}$$

- For discrete pixel values maximum/minimum can be used.

Edge-based techniques: ECR (III)

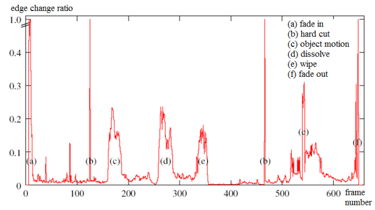
Detection of cuts

- One increased value
  → hard cut
- Several slightly increased values
  → soft cut
- Additional indicator for soft cuts:
  Ratio of incoming and outgoing edges:
  – Edges disappear in case of a fade out or in the first part of a dissolve ($p_{out(i)} > p_{in(j)}$),
  – The ratio is inverse in case of a fade in or in the second part of a dissolve

Edge-based techniques: ECR (IV)

Reliability

- High number of incorrect cuts caused by object motion



Quality evaluation (I)

Precision

$$P = \frac{C}{C+F} \in [0,1]$$

Recall

$$V = \frac{C}{C+M} \in [0,1].$$

- C: number of correctly detected cuts
- F: number of incorrect cuts
- M: missed cuts (not detected)
- All cuts are real cuts :
  → maximize precision
- All correct cuts are detected
  → maximize recall

Quality evaluation (II)

- The individual measurements (precision *or* recall) do not describe the quality of a cut detection technique
- F1 measurement:

$$F1 = 2 \cdot \frac{P * V}{P + V} \in [0,1] \quad \text{for} \quad P, V \neq 0.$$

Quality evaluation (III)

Example: detection of hard cuts

Approach	Precision	Recall	F1	Computational effort
Sum of absolute differences	85,2 %	82,7 %	83,9 %	0,86
Edge change ratio	76,1 %	86,5 %	81,0 %	7,78
Histogram	60,4 %	79,2 %	68,5 %	0,67
Average color	56,9 %	68,2 %	62,0 %	0,67
Contrast	55,7 %	68,9 %	61,6 %	0,76
Edge based contrast	13,3 %	23,5 %	16,9 %	0,75

(results based on 9 test videos)

Optimization for hard cuts (I)

Goal: Reduce high error rates in case of fast motion

- Hard cut if $D_{(i)} > D_{avg(i)} + T$

  dynamic threshold:

$$D_{avg}(i) = \frac{1}{N} \sum_{j=i-\frac{N}{2}, j \neq i}^{i+\frac{N}{2}} D(j)$$

Optimization for hard cuts (II)

Examples based on a dynamic threshold

Approach	Precision	Recall	F1
Sum of absolute differences	94,4 %	94,2 %	94,3 %
Edge change ratio	82,8 %	92,2 %	87,2 %
Histogram	81,4 %	89,0 %	85,0 %
Average color	74,1 %	76,6 %	75,3 %
Contrast	72,7 %	74,6 %	73,6 %
Edge based contrast	21,1 %	20,4 %	20,7 %

Detection of soft cuts (I)

Detection of soft cuts

- In most cases, it is not possible to distinguish between object / camera motion and soft cuts
  → significantly higher error rates

Approach based on contrast / edge contrast

1. Analyze interval: Values drop up to the middle of a dissolve due to the overlay of two frames; values increase in the second part of the dissolve.
2. Identify local minimum at the center of a dissolve
3. Validate that the difference between minimum and maximum value exceeds a threshold

Detection of soft cuts (II)

Approach based on ECR

$$M_{ECR}(i) = \sum_{j=i-\frac{N}{2}}^{i+\frac{N}{2}} ECR(j) - \max\left\{ECR(j) : j = i - \frac{N}{2} \dots i + \frac{N}{2}\right\}.$$

- Summarize ECR values within interval and subtract maximum ECR value
  → hard cuts are ignored this way

Detection of soft cuts (III)

Approach based on histograms

$$M_{HD}(i) = HD_N(i - \frac{N}{2}) - \max\left\{HD_1(j) : j = i - \frac{N}{2} \dots i + \frac{N}{2}\right\}.$$

- $HD_1(j)$: histogram difference of two adjacent frames j and $j + 1$
- $HD_N(i)$: histogram difference of image i and $i + N$

  → The value of the modified histogram difference M_{HD} is very high, only in case of a soft cuts

Detection of soft cuts (IV)

Dissolve

Approach	Precision	Recall	F1
Edge change ratio	45,0 %	43,1 %	44,0 %
Histogram	58,3 %	52,1 %	55,0 %
Contrast	54,2 %	59,2 %	56,6 %
Edge based contrast	46,1 %	37,9 %	41,6 %

Fade in / fade out

Approach	Precision	Recall	F1
Contrast	97,7 %	74,4 %	84,5 %
Edge based contrast	93,0 %	72,1 %	81,2 %

Questions?