Color Adaptation of Videos for Mobile Devices

Stephan Kopf, Fleming Lampi, Wolfgang Effelsberg
University of Mannheim
kopf@informatik.uni-mannheim.de

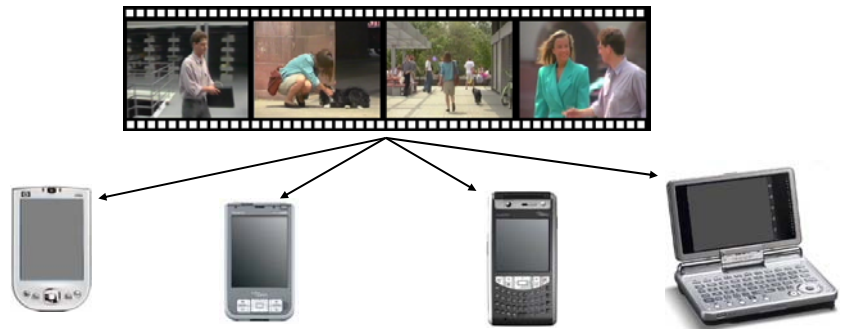
Outline

- Introduction
- Video adaptation approaches
- Adaptation for grayscale displays
  - Linear / non-linear transformation
  - Video adaptation
- Adaptation for binary (black and white) displays
  - Dithering algorithms
  - Texture-based adaptation
- Experimental results
- Conclusion and outlook

Introduction

Goal

- Visualize recorded videos on arbitrary devices
- Preserve the semantic content of the videos



Introduction (cont.)

- Specific features of a device should be considered for the adaptation
  - resolution and color depth of the display
  - memory
  - CPU
  - software
  - available network capacity
- Automatic adaptation algorithms are required (manual adaptation is not possible)

Video adaptation approaches

Classification of video adaptation approaches

- Location: **server**, proxy, client
- Technical features of the devices: **hardware**, software, available network capacity
- Type of adaptation: static, **dynamic**
- Transcoding: resolution, **color**, bit rate, frame rate

→ Now, we focus on color adaptation of videos.

Adaptation for grayscale displays

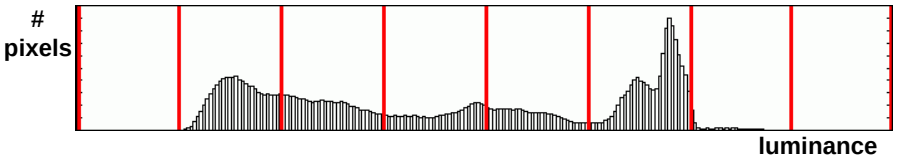
- Large regions with identical colors appear when reducing the color depth:

- A major challenge is the adaptation for monochrome (black and white) displays.

Adaptation based on linear transformations

$$L_{lin}(i) = \lfloor \frac{N_C}{256} \cdot i \rfloor \in [0, N_C - 1].$$

N_C: number of different luminance values



- Many details are lost due to the equidistant intervals.

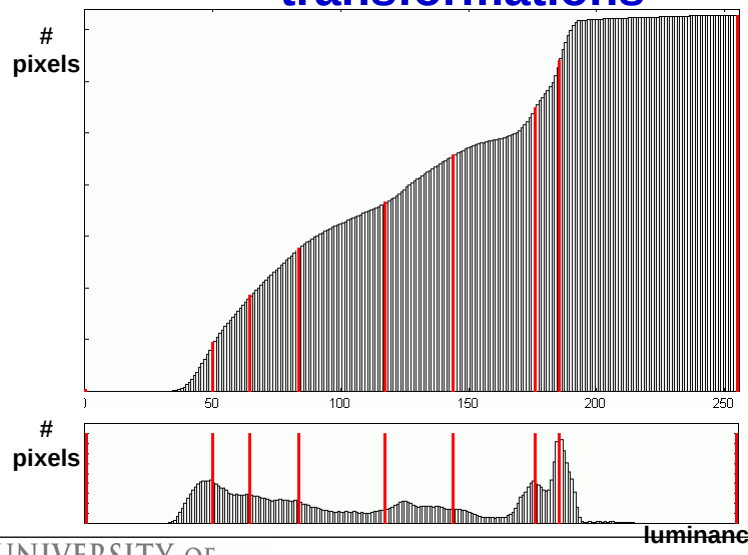
Adaptation based on non-linear transformations

- Use variable sized intervals derived from the distribution of luminance values

$$L_{var}(i) = \lfloor \frac{N_C}{S_X \cdot S_Y + 1} \cdot H_{kum}(i) \rfloor \in [0, N_C - 1].$$

H_{kum}: cumulated histogram

Adaptation based on non-linear transformations



Adaptation based on non-linear transformations (cont.)

- Variable-sized intervals change the average luminance in very dark or bright scenes.
- Combine the linear and non-linear adaptation:

$$L_w(i) = \lfloor \alpha \cdot L_{lin}(i) + (1-\alpha) \cdot L_{var}(i) \rfloor \ \in \ [0, N_C - 1].$$

Factor α weights both approaches.

Video adaptation (frame-based)

Video adaptation

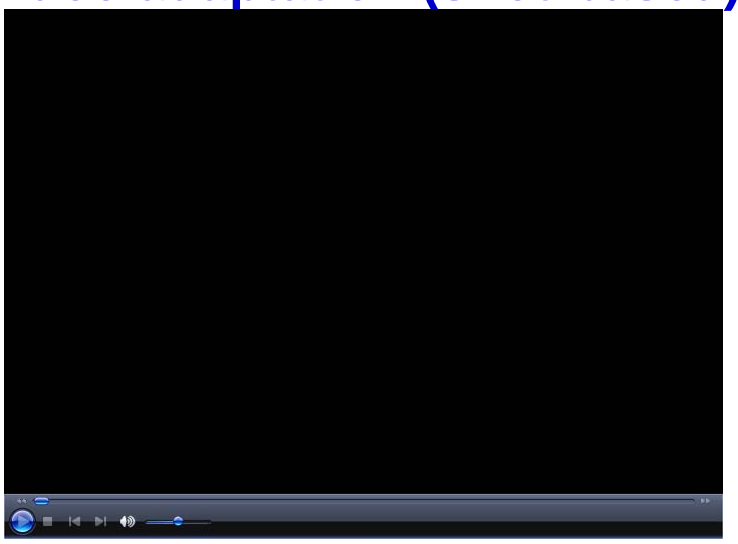
Problems

- The parameters and transformation should not change within a camera shot.
- The non-linear transformation leads to noisy images in case of very dark or bright shots.

Solution

- Calculate aggregated cumulated histogram for all frames of a shot.
- Discard very dark or bright frames.
- Derive suitable intervals for each shot.

Video adaptation (shot-based)

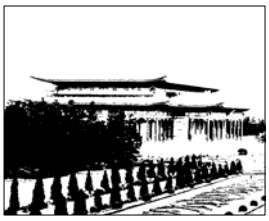


Adaptation for binary displays

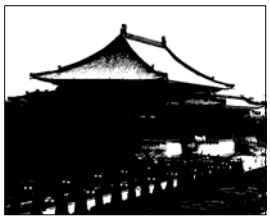
Idea

- Offset printing: few colors are used and fine dots are printed next to each other.
- The human eye combines and merges luminance values of adjacent pixels.
- Floyd/Steinberg dithering algorithm:
  - A pixel is set to the most similar color
  - The error is shifted to adjacent pixels (error diffusion).

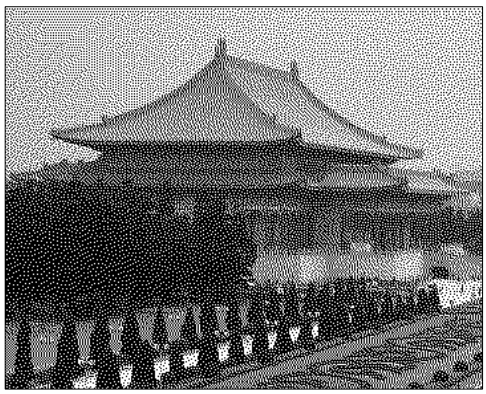
Adaptation for binary displays (cont.)



threshold: 90



threshold: 130



Floyd/Steinberg

Adaptation for binary displays (cont.)

- Dithering works well for images, but it is **not applicable** for video sequences.

→The diffusion of the error to adjacent pixels creates a lot of noise.

→The values of many pixels change between two frames.

→ The quality of the adapted video is very low.

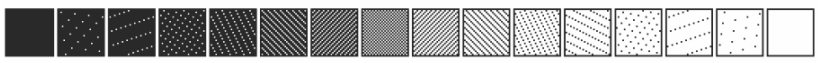
Video adaptation (Floyd/Steinberg)

Texture-based adaptation

- We use binary textures to overlay the image

$$I_t(x,y) = \begin{cases} 0 & [(x + S_X \cdot y) \text{ MOD } (T_B + T_W)] < T_B, \\ 1 & \text{else.} \end{cases}$$

T_B, T_W define the ratio of black and white pixels

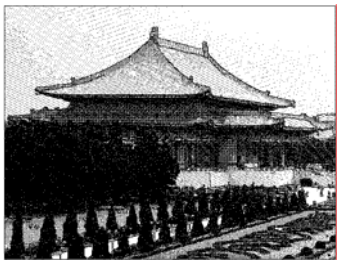


$T_B >> T_W$ $T_B << T_W$

- Textures which represent similar luminance values should be easily distinguishable.

Texture-based adaptation (cont.)

1. Create a grayscale image with N_C = 16 different luminance values based on cumulated histograms.
2. Replace each value with a texture.

→ strong edges are lost

3. Emphasize strong edges.

Experimental results

Conclusion and Outlook

- We presented new algorithms to adapt the color depth of videos.
- **Cumulated histograms** aggregated **on shot level** lead to a good adaptation of grayscale videos.
- A major deficiency of the image based dithering algorithms is the great amount of noise in videos.
- A good solution for the adaptation of binary videos is to use **textures** and emphasize strong **edges**.
- We plan to integrate different adaptation algorithms (color, screen resolution, bandwidth) to enable the playback of videos on all different types of mobile devices.

Video adaptation homepage:

http://www.informatik.uni-mannheim.de/pi4/projects/moca

kopf@informatik.uni-mannheim.de