

# Color Adaptation of Videos for Mobile Devices

Stephan Kopf, Wolfgang Effelsberg

Department of Computer Science IV, University of Mannheim, Mannheim, Germany

## ABSTRACT

A large number of recorded videos cannot be viewed on mobile devices (e.g., PDAs or mobile phones) due to inappropriate screen resolutions or color depths of the displays. Recently, automatic transcoding algorithms have been introduced which facilitate the playback of previously recorded videos on new devices. One major challenge of transcoding is the preservation of the semantic content of the videos. Although much work was done on the adaptation of the image resolution, color adaptation of videos has not been addressed in detail before. In this paper, we present a novel color adaptation algorithm for videos which preserves the semantics. In our approach, the *color depth* of a video is adapted to facilitate the playback of videos on mobile devices which support only a limited number of different colors. We analyze our adaptation approach in the experimental results, visualize adapted keyframes and illustrate, that we obtain a better quality and are able to recognize much more details with our approach.

**Keywords:** video adaptation, color adaptation, content repurposing, transcoding

## 1. INTRODUCTION

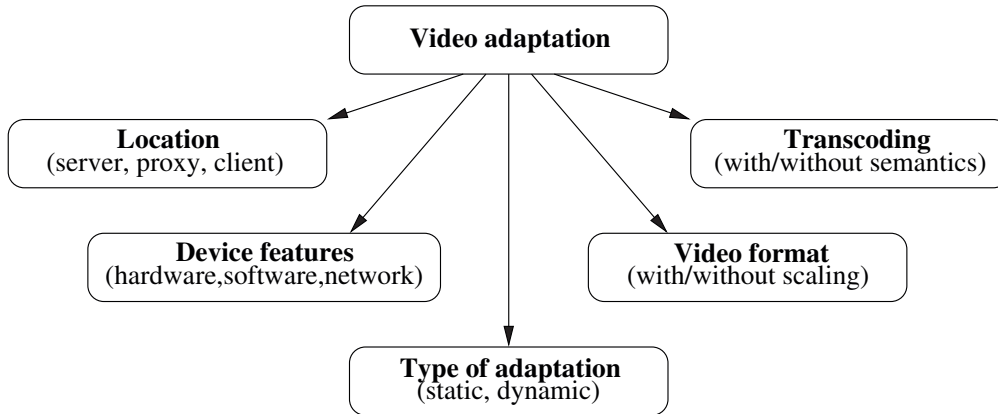
Videos are no longer limited to television or personal computers due to the technological progress in the last years. Nowadays, many different devices support the playback of videos. A major challenge is the *heterogeneity* of mobile devices, especially the different device classes (notebook, Handheld-PC, PDA, or mobile phone) with many different features. Major characteristics for the different device classes are the resolution and color depth of the display, the memory, the CPU, and the software to decode and visualize the video. A video is usually transferred in real-time to the mobile device. Therefore, the available network capacity is a potential bottleneck, too.

A significant reduction of the quality of the video is typical if it is played on a mobile device. Basic problems have to be solved before existing videos can be presented in an acceptable way. The specific features of a mobile device must be considered for the adaptation. A manual specification of adaptation parameters is unfeasible due to the large number of videos, device classes and combinations of hardware, software and network capacities. Therefore, an *automatic video adaptation technique* facilitates the playback of videos especially for mobile devices. An important goal is the preservation of the semantic information in the adapted videos. Essential parameters for the adaptation of videos are the bit rate, the color depth, the image resolution and the frame rate. In this paper, we focus on the adaptation of the color depth and do not consider the other parameters. A change of the color depth is not trivial if we want to preserve the visual content in the adapted video.

In this paper, we present a new approach to adapt the color resolution of videos for mobile devices. The following section gives an extensive overview of previous work in the context of video adaptation. Section 3 illustrates our new approach for the *adaptation of the color depth*. The luminance distribution is analyzed for each shot to adapt videos for grayscale displays, and edge information is combined with textures to support binary (black-and-white) displays which are still quite common for mobile devices, especially mobile phones. In Section 3.4, we present experimental results and conclude the paper in Section 4.

## 2. ADAPTATION OF VIDEOS

Approaches for the adaptation of videos can be classified according to Figure 1.<sup>1</sup> A typical classification scheme distinguishes between *server*-,<sup>2-4</sup> *proxy*-<sup>5,6</sup> or *client*-based<sup>7</sup> adaptations. A bottleneck in the server-based approach is the CPU, especially if many clients connect to the server simultaneously. On the other hand, the client-based approach is unsuitable for videos due to the large amount of data which must all be sent to the client for adaptation.<sup>8</sup>



**Figure 1.** Classification of video adaptation approaches

*Technical features* of a mobile device define the specific parameters for the video adaptation.<sup>9</sup> Another classification criterion is the *type of adaptation*. If the complexity of the algorithms is not too high an adapted videos is computed in real-time (dynamic adaptation). Otherwise, different adapted static versions of a video are calculated offline and stored on the server for later usage without any further processing.

*Scaling* is supported in several video standards (e.g., MPEG-2), where different quality levels are stored in one video stream. The *base layer* uses a minimum amount of network and CPU resources. Additional *enhancement layers* improve the quality of the videos step-by-step.

A *transcoding* algorithm modifies the format-specific parameters of a video, for example, the color depth, resolution, frame rate, or bit rate. Semantic transcoding considers the visual content of the video and tries to preserve the semantic content in the adapted video.<sup>10</sup>

Our adaptation approach is server-based and considers the color depth of mobile devices. A dynamic (real-time) adaptation algorithm is used. Our algorithms focus on semantic transcoding to preserve the visual content of videos.

## 2.1. Adaptation and standardization

The semantic description of the content of videos is part of the *MPEG-7* and *MPEG-21* standards,<sup>11,12</sup> and metadata is specified for the personalization and the adaptation of videos.<sup>13,14</sup> *MPEG-7* encompasses a description language for easy exchange of multimedia documents. Additionally, the network-based access to multimedia documents from arbitrary devices is supported (*universal multimedia access*).<sup>3,15,16</sup> Rules to transcode videos, a user history and individual user preferences (*user preference description*) can be stored and used for the automatic adaptation.

Additional metadata is integrated into the *MPEG-21* standard. A description of the user's device (*usage environment description*) is available to characterize the display, the hardware and the configuration of a mobile device.<sup>17</sup> Additional techniques like the modeling of user queries and user preferences are part of *MPEG-21*. Individual adaptation algorithms can be defined in *MPEG-21* for digital elements (*digital item adaptation*).<sup>18</sup>

## 2.2. Video adaptation algorithms

Many adaptation algorithms in the context of image and audio adaptation are also suitable for videos. The goal of *image adaptation* is the adjustment of image parameters to the physical features of the display, especially the color depth and resolution.<sup>19,20</sup> Relevant semantic content is lost by decreasing the color depth. This problem is amplified for black-and-white displays.<sup>21</sup>

An interesting approach is to identify relevant objects (*attention objects*), which should be recognizable in the adapted image.<sup>22–25</sup> Important objects combined with an analysis of the color distribution, the contrast and the orientation of edges<sup>26</sup> are suitable to define *regions of interest*.<sup>27,28</sup>

Two different approaches are considered to *adapt audio signals*. The first one changes the format-specific parameters of the audio signal, e.g., the coded frequencies or the audio quality. On the other hand, the audio signal can be transformed into another kind of representation. Speech recognition is especially relevant in this context and was frequently used to analyze videos, for instance to index newsreels.<sup>29,30</sup>

Many efficient transcoding algorithms were developed for the *adaptation of videos*.<sup>31</sup> Missing software decoders or insufficient hardware support on mobile devices necessitate a change of the video codec.<sup>32–35</sup> The transcoding of videos should keep the computational effort as low as possible and reuse data from complex calculations like motion vectors.<sup>36</sup> Most systems propose efficient algorithms to transcode videos but do not consider the semantic content of the videos.<sup>37–39</sup> Other approaches identify the semantic content in videos and emphasize important details in the adapted video.<sup>39–41</sup> Objects and special events indicate regions of interest.<sup>42–45</sup>

The network capacity is not always sufficient to transmit a video to a mobile device. In this case, a different kind of presentation should be used. For instance, a video could be presented as a slide show which visualizes selected key frames and combines them with the audio track.<sup>46</sup> A second challenge is the unreliability of wireless networks, especially if the user is on the move (e.g., in cars or trains). One solution to this problem could be to send the video to the receiver in advance. This leads to the next challenge: handling the limited storage capacities of mobile devices. A video summary that preserves the essential content of the much longer video is a suitable solution.<sup>47–49</sup>

Although much work was done on the adaptation of videos most approaches focus on efficient algorithms for transcoding. Many publications do not consider the essential problem to preserve the *semantic content*. Additionally, only a very limited amount of work was done on the video adaptation of the color depth.

### 3. ADAPTATION OF THE COLOR DEPTH

When reducing the color depth of a grayscale image large regions with identical colors appear, and it becomes much more difficult to recognize the visual content. A major challenge is the adaptation of videos for monochrome displays where all pixels are represented with two different luminance values. In this section, we present an adaptation algorithm to generate videos with a limited number of different grayscale values first, and focus on the adaptation of videos for binary displays in a second step.

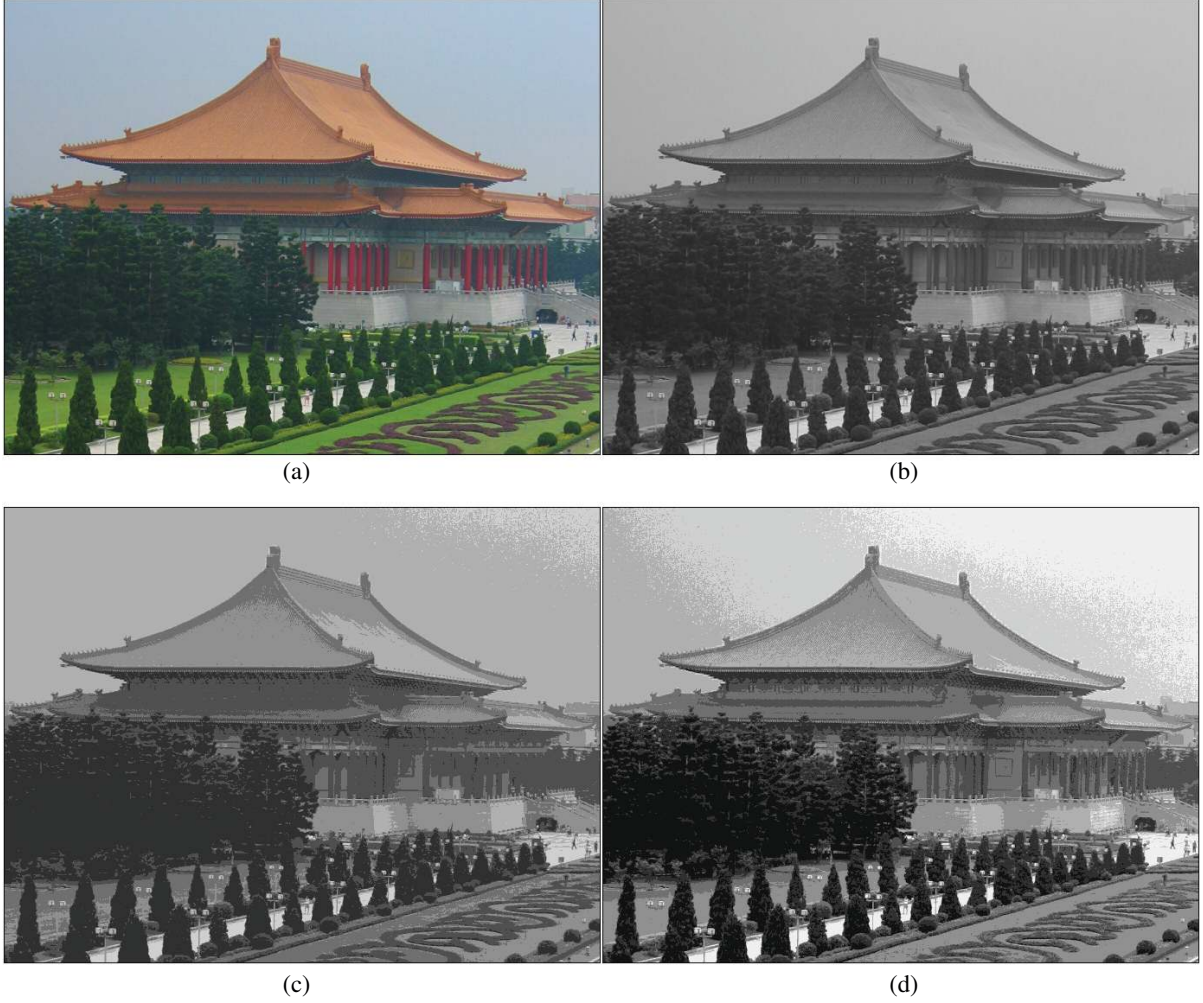
#### 3.1. Adaptation for grayscale displays

The conversion from color to grayscale is done without any computational effort because most video compression standards store luminance and color values separately. However, details are lost when the number of different luminance values is reduced. The histograms of most images illustrate that the distribution among the luminance values is not uniform and that a large percentage of the pixels is located within small intervals. A linear transformation of luminance values, which uses equal-sized intervals for the mapping, generates large monochrome regions in images. The full contrast of the display is not used, and many details are lost.

We now present a new approach to reduce the color depth from 256 to  $N_C$  different luminance values. We present some examples based on grayscale images with  $N_C = 8$  different luminance values. The *linear transformation* uses equal-sized intervals with a size of  $\frac{256}{N_C}$ . All luminance values within an interval are mapped to the same value:

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

All pixels in the source image with the luminance  $i$  are transformed to the new value  $L_{lin}(i)$ . Figure 2 (c) exemplifies that fine structures and details are lost in the adapted image in case of  $N_C = 8$  different luminance values. A *variable interval size* derived from the distribution of the luminance values in the source image



**Figure 2.** Transformation of a color image (a) into grayscale images with 256 (b) and eight luminance values using linear (c) and non-linear adaptation (d).

significantly improves the quality of the adapted image. Cumulated histograms  $H_{kum}(i)$  are used to define the non-linear transformation of the luminance values:

$$L_{var}(i) = \lfloor \frac{N_C}{S_X \cdot S_Y + 1} \cdot H_{kum}(i) \rfloor \in [0, N_C - 1]. \quad (2)$$

The width  $S_X$  and height  $S_Y$  of the image normalize the cumulated histogram. The transformation of the luminance  $i$  to the new value  $L_{var}(i)$  depends on the distribution of the values of the histogram. Figure 2 (d) illustrates that much more details are discernible if a variable interval size is applied.

A major disadvantage of the nonlinear adaptation is the change of the average luminance in very dark or very bright images. The parameter  $\alpha \in [0, 1]$  is used to combine the linear and non-linear adaptation and to limit the change of the average luminance of an image:

$$L_w(i) = \lfloor \alpha \cdot L_{lin}(i) + (1 - \alpha) \cdot L_{var}(i) \rfloor \in [0, N_C - 1]. \quad (3)$$

### 3.2. Adaptation for videos

In the previous section, we focused on the adaptation of images. We consider an extension for the *adaptation of videos* in the next step. It is important, that the parameters and the transformation is unchanged within a camera shot. Otherwise, even without camera or object motion many changes of pixels at the same image position would occur. This is the case, e.g., if the average luminance value changes within a shot. This effect is very strong in case of fades and the nonlinear transformation leads to very noisy images, although this effect can be reduced by the parameter  $\alpha$  of Equation 3. To gain better results, we analyze the cumulated histograms for each frame and calculate one aggregated cumulated histogram for all frames of a shot. The histogram describes the distribution of the luminance values of one shot; and suitable parameters  $L_{var}(i)$  can be derived for this shot. We do not consider very dark or bright frames (e.g., fades or flashlights) to avoid significant modifications of the cumulated histograms.

### 3.3. Adaptation for binary displays

The *adaptation for binary displays* is considered in a second step. The visualization of an image with a limited number of colors or luminance values is a well-known problem of printing. The technique is known in offset printing, where the printer supports few colors only (no half-tones). The image is rastered and fine dots are printed next or on top of each other. An image color is mapped based on a color palette. In case of binary images, the naive approach is to define a threshold and compare each pixel with this threshold. Two different thresholds are applied in the examples of Figure 3 (a) and (b). Much information is lost in both images and the naive approach does not work very well.

The human eye combines and merges the luminance values of adjacent pixels, so that a much better approach to visualize an image is possible. In 1975, Floyd and Steinberg published an algorithm which reduces the visible error significantly.<sup>50</sup> The image is transformed pixel by pixel: Starting with the upper left corner, the pixel is transformed to the most similar color or luminance value. The error caused by the new pixel value is shifted to adjacent pixels (*error diffusion*). Errors are shifted to the right pixel and to pixels in the following line. Figure 3 (c) visualizes, that the quality of the adapted image based on the Floyd/Steinberg algorithm is much better.

Although Floyd/Steinberg works very well for images, the idea of error diffusion is *not applicable for video sequences*. The diffusion of the error to adjacent pixels creates a large amount of noise in videos and many pixels change their value from frame to frame. Therefore, the quality is very low if videos are adapted with the Floyd/Steinberg algorithm.

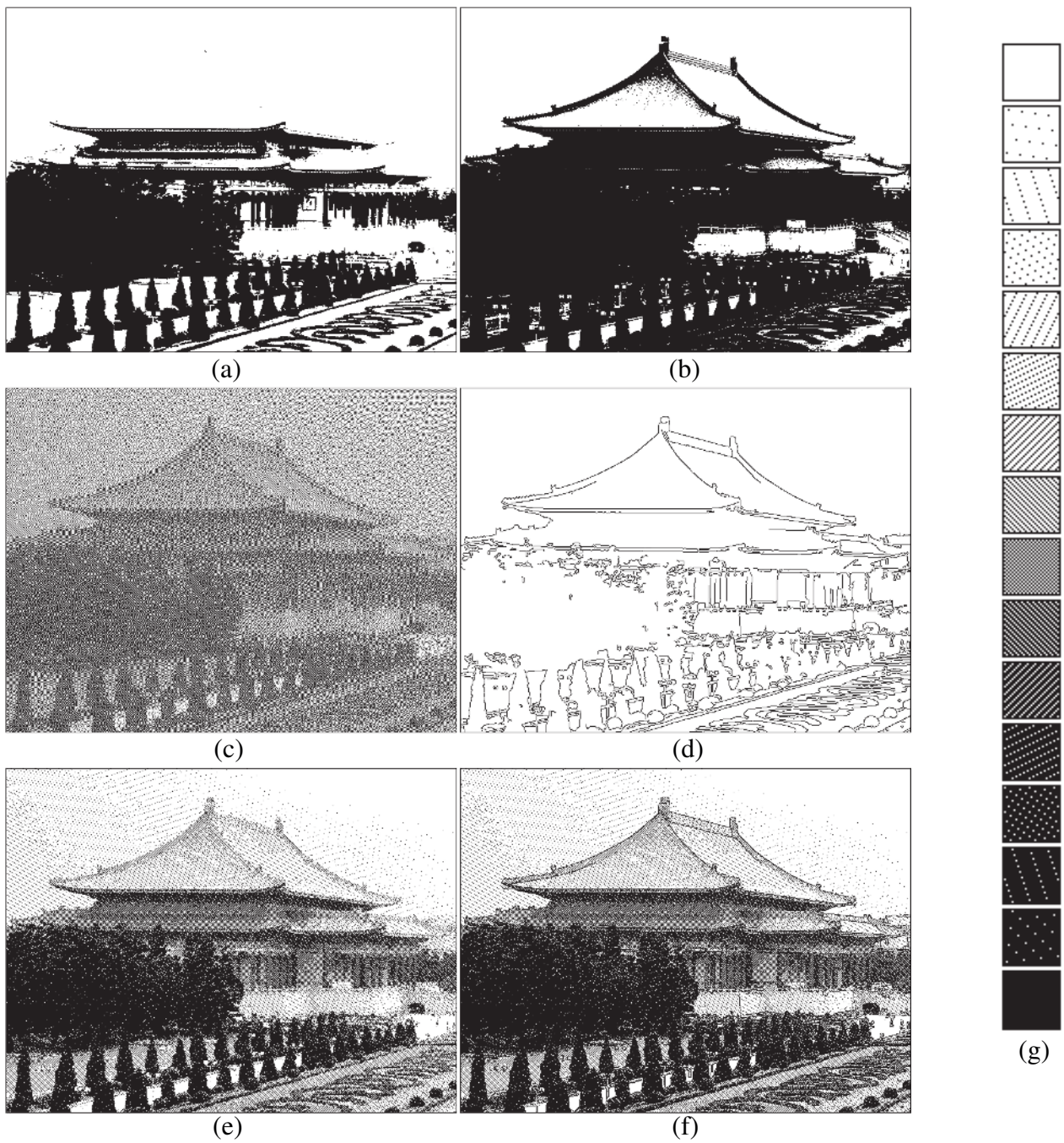
An alternative and more stable visualization of binary images uses edges. The edge image in Figure 3 (d) depicts many details but the objects and coherent regions in the image cannot be recognized very well.

More details are visible if *binary textures* are used to overlay the image. In the following example 16 textures  $I_t(x, y)$  are defined to substitute the pixel of the grayscale image:

$$I_t(x, y) = \begin{cases} 0 & [(x + S_X \cdot y) \text{ MOD } (T_B + T_W)] < T_B, \\ 1 & \text{else.} \end{cases} \quad (4)$$

The values  $T_B$  and  $T_W$  define the ratio of black and white pixels of the texture. The value of  $T_B$  is significantly larger than  $T_W$  first and the ratio is shifted to bright pixels more and more. The values  $T_B$  and  $T_W$  are manually selected, so that the textures which represent similar luminance values are easily distinguishable. Coherent regions are recognized much easier. Figure 3 (g) depicts 16 easily distinguishable textures derived from Equation 4.

Before we can apply these textures, a grayscale image with  $N_C = 16$  different luminance values is constructed with cumulated histograms, and each value is substituted with pixels from the corresponding texture. In case of videos, the adaptation is based on the cumulated histograms of the shots. Nevertheless, the differences between two adjacent regions are quite low in some cases. This leads to good results for gradual transitions (e.g., the sky in Figure 3 (e)) but strong edges of the image are lost. Hence, significant edge pixels are emphasized by combining the edge image with the texture image (see Figure 3 (f)). Most details are visible in comparison to the other binary images.



**Figure 3.** Transformation of a color image into a binary image applying fixed thresholds of 90 (a) and 130 (b). The quality of dithering algorithms (c) is very good for images; but they are not applicable for videos. The edge image (d) emphasizes object borders. The semantic adaptation with textures (g) is suitable for videos and visualizes much more details (e). Object borders are highlighted by overlaying the adapted image with its edge image (f).

### 3.4. Experimental Results

We present some adapted binary images from video sequences in the experimental results. Sample videos and a video demo are available on the Web.<sup>51,52</sup> Figure 4 depicts keyframes from two videos which are transformed into the binary format. The adaptation is visualized for the naive approach with a fixed threshold in Figure 4 (b) and our new method in (c). An optimal threshold for each video was selected manually for the naive approach. Nevertheless, large parts of the images do not depict any relevant information. A decrease of the image quality is expected if the threshold is calculated automatically.

The adapted images based on our new algorithm are presented in Figure 4 (c). The overlay of binary textures combined with the edge image visualizes much more details. The adaptation uses intervals of variable size derived from calculated histograms based on shots. Therefore, a manual definition of the thresholds is not required.

A second example visualizes keyframes of a second adapted video. Figure 5 compares 3 bit and binary keyframes of this video. The content can be recognized very well, although only 8 different grayscale values are used in Figure 5 (b). Even the binary images in Figure 5 (c) represent the content very well.

## 4. CONCLUSION AND OUTLOOK

We presented a novel algorithm for the adaptation of the color depth of videos. In case of grayscale images, we used cumulated histograms to define a suitable mapping of the color values and extended this approach by aggregating the luminance values for each shot. The overlay of textures in combination with strong edges works very well for the adaptation into binary images.

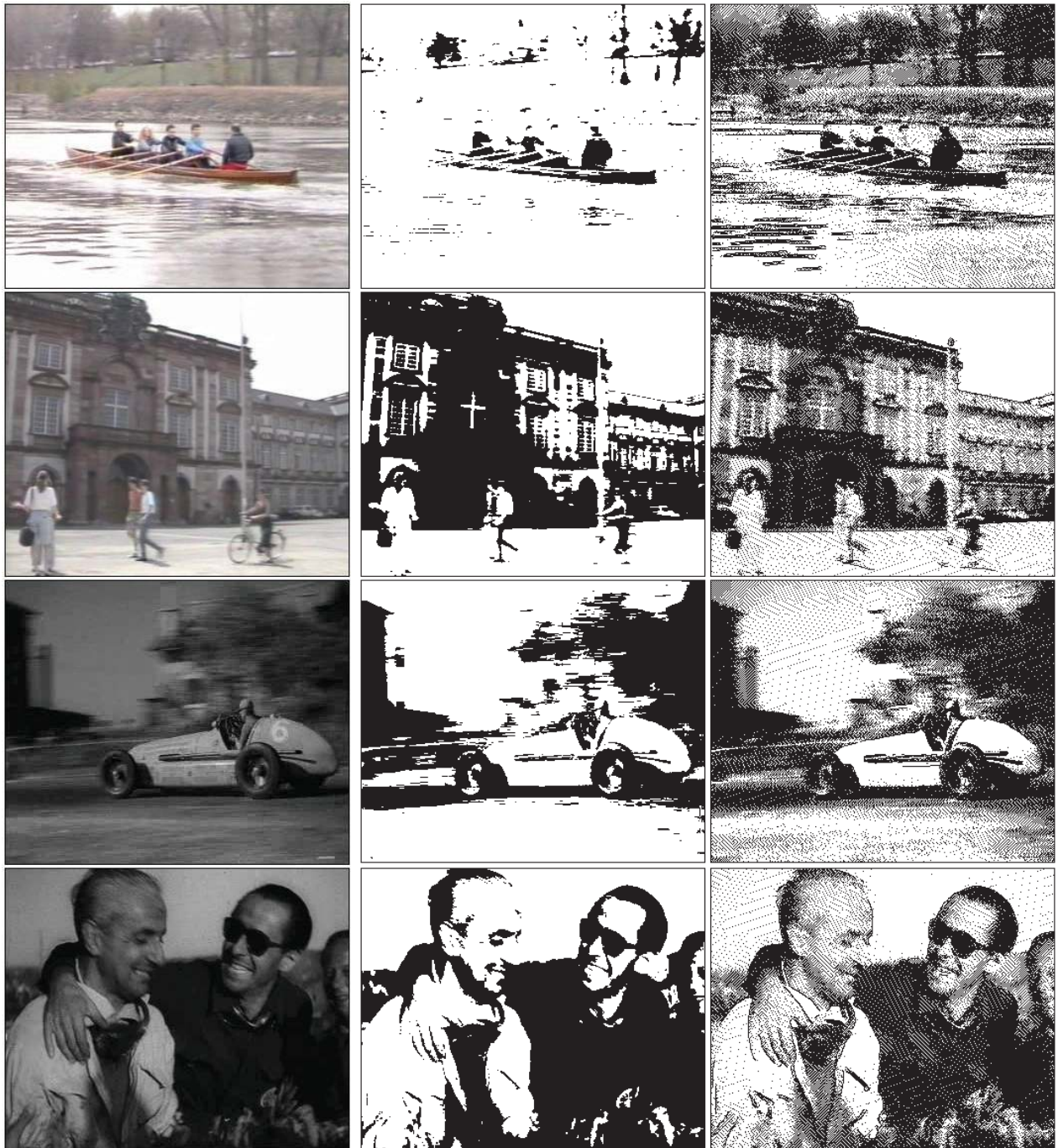
The color adaptation of videos is a first step to automatically create adapted videos which can be visualized on mobile devices, especially mobile phones. The adaptation of other parameters of videos is very important, too, like the modification of the image resolution. We are currently working on the combination of different adaptation approaches. The long-term goal is to provide tools for the automatic adaptation of videos, so that recorded videos can be presented in high quality on heterogeneous devices.

## 5. ACKNOWLEDGMENTS

This work was supported by grants from the Deutsche Forschungsgemeinschaft (DFG).

## REFERENCES

1. H. Lie and J. Saarela, "Multipurpose web publishing using HTML, XML and CSS," in *Communications of the ACM*, **42(10)**, pp. 95–101, ACM Press, October 1999.
2. M. Hossain, A. Rahman, and A. Saddik, "A framework for repurposing multimedia content," in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, pp. 971–974, IEEE Computer Society Press, May 2004.
3. R. Mohan, J. Smith, and C. Li, "Adapting multimedia internet content for universal access," in *IEEE Transactions on Multimedia*, **1(1)**, pp. 104–114, IEEE Computer Society Press, March 1999.
4. B. Noble, M. Satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, , and K. R. Walker, "Agile application-aware adaptation for mobility," in *Proceedings of the 16th Symposium on Operating System Principles*, pp. 276–287, 1997.
5. R. Han, P. Bhagwat, R. LaMayre, T. Mummert, V. Perret, and J. Rubas, "Dynamic adaptation in an image transcoding proxy for mobile www browsing," in *IEEE Personal Communication*, **5(6)**, pp. 8–17, IEEE Computer Society Press, 1998.
6. W. Lum and F. Lau, "A context-aware decision engine for content adaptation," in *IEEE Pervasive Computing*, **1(3)**, pp. 41–49, IEEE Computer Society Press, July 2002.
7. Z. Lei and N. D. Georganas, "Context-based media adaptation in pervasive computing," in *Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering*, **2**, pp. 913–918, IEEE Computer Society Press, May 2001.



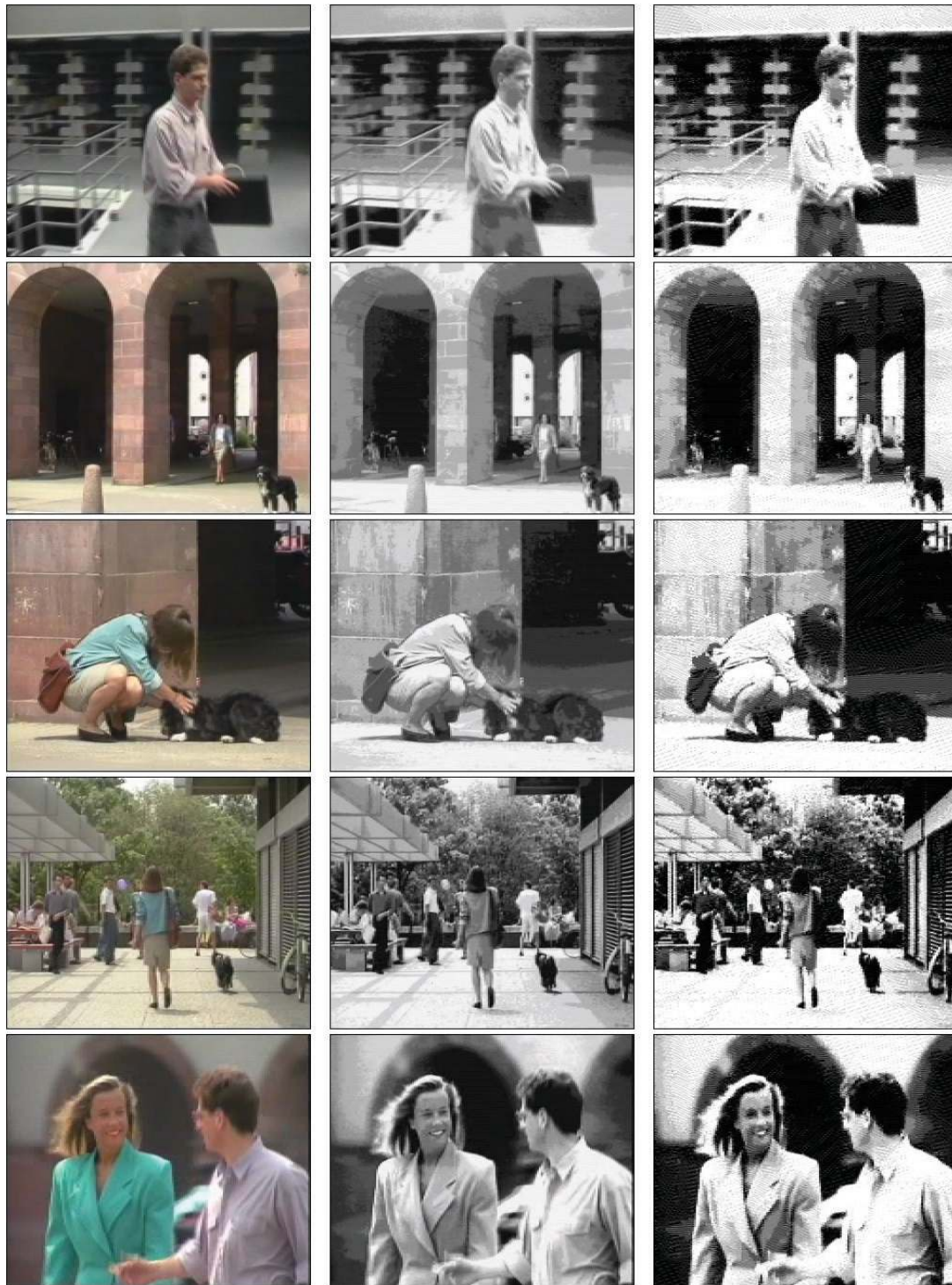
(a)

(b)

(c)

**Figure 4.** Examples of an adapted color video (top) and a historical video (bottom): Selected keyframes of the original video (a), the adapted binary videos generated with an optimal threshold (b) and the semantically adapted video based on overlaid textures (c).





(a)

(b)

(c)

**Figure 5.** Examples of selected keyframes of a video: Original video (a), video based on 3 bit grayscale images (b), and video in binary format (c).

8. V. Cardellini, P. Yu, and Y. Huang, "Collaborative proxy system for distributed web content transcoding," in *Proceedings of 9th International ACM Conference on Information and Knowledge Management*, pp. 520–527, ACM Press, November 2000.
9. A. Fox, S. Gribble, Y. Chawathe, and E. Brewer, "Adapting to network and client variation using infrastructural proxies: Lessons and perspectives," in *IEEE Personal Communication*, **5(4)**, pp. 10–19, IEEE Computer Society Press, 1998.
10. Z. Obrenovic, D. Starcevic, and B. Selic, "A model-driven approach to content repurposing," in *IEEE Multimedia*, **11(1)**, pp. 62–71, IEEE Computer Society Press, January 2004.
11. ISO/IEC, "Information technology – multimedia content description interface (MPEG-7) – part 8: Extraction and use of MPEG-7 descriptions," Tech. Rep. TR 15938-8, ISO/IEC, 2002.
12. ISO/IEC, "Information technology – multimedia framework (mpeg-21) – part 1: Vision, technologies and strategy," Tech. Rep. TR 21000-1, ISO/IEC, 2004.
13. B. Tseng, C.-Y. Lin, and J. R. Smith, "Using MPEG-7 and MPEG-21 for personalizing video," in *IEEE Multimedia*, **11(1)**, pp. 42–52, IEEE Computer Society Press, January 2004.
14. A. Vetro, "MPEG-21 digital item adaptation: Enabling universal multimedia access," in *IEEE Multimedia*, **11(1)**, pp. 84–87, IEEE Computer Society Press, January 2004.
15. P. Beek, J. R. Smith, T. Ebrahimi, T. Suzuki, and J. Askelof, "Metadata-driven multimedia access," in *IEEE Signal Processing Magazine*, **20(2)**, pp. 40–52, IEEE Computer Society Press, March 2003.
16. A. Vetro, T. Christopoulos, and T. Ebrahimi, "Special issue on universal multimedia access," in *IEEE Signal Processing Magazine*, **20(2)**, pp. 69–79, IEEE Computer Society Press, March 2003.
17. I. Nurnett, "MPEG-21: Goals and achievements," in *IEEE Multimedia*, **10(6)**, pp. 60–70, IEEE Computer Society Press, October 2003.
18. ISO/IEC, "MPEG-21 multimedia framework – part 7: Digital item adaptation (final committee draft)," Tech. Rep. N 5845, ISO/IEC, 2003.
19. D. Kasik, "Strategies for consistent image partitioning," in *IEEE Multimedia*, **11(1)**, pp. 32–41, IEEE Computer Society Press, January 2004.
20. T. Rist and P. Brandmeir, "Customizing graphics for tiny displays of mobile devices," in *Proceedings of 3rd International Workshop on Human Computer Interaction with Mobile Devices*, pp. 1–4, September 2001.
21. T. Rist and P. Brandmeier, "Customizing graphics for tiny displays of mobile devices," in *Personal and Ubiquitous Computing*, **6(4)**, pp. 260–268, Springer, 2002.
22. L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou, "A visual attention model for adapting images on small displays," in *ACM Multimedia Systems Journal*, **9(4)**, pp. 353–364, ACM Press, 2003.
23. X. Fan, X. Xie, W. Ma, H. Zhang, and H. Zhou, "Visual attention based image browsing on mobile devices," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, **1**, pp. 53–56, IEEE Computer Society Press, July 2003.
24. S. Kopf, T. Haenselmann, D. Farin, and W. Effelsberg, "Automatic generation of summaries for the web," in *Proceedings of IS&T/SPIE conference on Storage and Retrieval for Media Databases*, **5307**, pp. 417–428, January 2004.
25. S. Kopf, F. Lampi, T. King, and W. Effelsberg, "Automatic scaling and cropping of videos for devices with limited screen resolution," in *Proceedings of the 14th ACM international conference on Multimedia*, pp. 957–958, ACM Press, October 2006.
26. L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20(11)**, pp. 1254–1259, IEEE Computer Society Press, November 1998.
27. J. Hu and A. Bagga, "Categorizing images in web documents," in *IEEE Multimedia*, **11(1)**, pp. 22–30, IEEE Computer Society Press, January 2004.
28. N. Rowe, "Content repurposing for small devices," in *Encyclopedia of Multimedia Technology and Networking (Volume I)*, M. Pagani, ed., **1**, pp. 110–115, The Idea Group, Hershey, PA, USA, April 2005.
29. J. Gauvain, L. Lamel, and G. Adda, "Transcribing broadcast news for audio and video indexing," in *Communications of the ACM*, **Vol. 43(2)**, pp. 64–70, ACM Press, February 2000.

30. S. Srinivasan, D. Petkovic, and D. Ponceleon, "Towards robust features for classifying audio in the cuevideo system," in *Proceedings of the ACM international conference on Multimedia (Part 1)*, pp. 393–400, ACM Press, 1999.
31. A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques. an overview," in *IEEE Signal Processing Magazine*, **20(2)**, pp. 18–29, IEEE Computer Society Press, March 2003.
32. B. Bai and J. Harms, "A multiview video transcoder," in *Proceedings of the 13th annual ACM international conference on Multimedia*, pp. 503–506, ACM Press, 2005.
33. N. Bjrk and C. Christopoulos, "Video transcoding for universal multimedia access," in *Proceedings of the 2000 ACM workshops on Multimedia*, pp. 75–79, ACM Press, 2000.
34. K. Curran and S. Annesley, "Transcoding media for bandwidth constrained mobile devices," in *International Journal of Network Management*, **15(2)**, pp. 75–88, John Wiley & Sons, Inc., March 2005.
35. Z. Lei and N. D. Georganas, "Rate adaptation transcoding for precoded video streams," in *Proceedings of the 10th ACM international conference on Multimedia*, pp. 127–136, ACM Press, 2002.
36. T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to lower spatio-temporal resolution and different encoding formats," in *IEEE Transactions on Multimedia*, **2(2)**, pp. 101–110, IEEE Computer Society Press, June 2000.
37. R. Hjelsvold, S. Vdaygiri, and Y. Leaute, "Web-based personalization and management of interactive video," in *Proceedings of the 10th international conference on World Wide Web*, pp. 129–139, 2001.
38. S. Nepal and U. Srinivasan, "DAVE: A system for quality driven adaptive video delivery," in *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, pp. 223–230, ACM Press, 2003.
39. O. Steiger, T. Ebrahimi, and D. Sanjuan, "MPEG-based personalized content delivery," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, **3**, pp. 45–48, IEEE Computer Society Press, September 2003.
40. M. Bertini, R. Cucchiara, A. Bimbo, and A. Prati, "An integrated framework for semantic annotation and adaptation," in *Multimedia Tools and Applications*, **26(3)**, pp. 345–363, Springer Science & Business Media B.V., August 2005.
41. B.-L. Yeo and M. Yeung, "Retrieving and visualizing video," in *Communications of the ACM*, **40(12)**, pp. 43–52, ACM Press, December 1997.
42. J.-G. Kim, Y. Wang, and S.-F. Chang, "Content-adaptive utility-based video adaptation," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, pp. 281–284, IEEE Computer Society Press, July 2003.
43. B. Tseng and C. Lin, "Personalized video summary using visual semantic annotations and automatic speech transcriptions," in *IEEE Workshop on Multimedia Signal Processing*, pp. 5–8, IEEE Computer Society Press, December 2002.
44. B. Tseng and J. Smith, "Hierarchical video summarization based on context clustering," in *Proceedings of IS&T/SPIE conference on Internet Multimedia Management Systems IV*, pp. 14–25, November 2003.
45. A. Vetro, T. Haga, K. Sumi, and H. Sun, "Object-based coding for long-term archive of surveillance video," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, **2**, pp. 417–420, IEEE Computer Society Press, 2003.
46. M. Yeung, "Video browsing using clustering and scene transitions on compressed sequences," in *Proceedings of IS&T/SPIE conference on Multimedia Computing and Networking*, **2417**, pp. 399–413, 1995.
47. Y. Gong and X. Liu, "Video summarization using singular value decomposition," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, **2**, pp. 174–180, IEEE Computer Society Press, 2000.
48. M. Smith and T. Kanade, "Video skimming for quick browsing based on audio and image characterization," Tech. Rep. CMU-CS-95-186, Carnegie Mellon University, 1995.
49. H. Sundaram and S. Chang, "Determining computable scenes in films and their structures using audio-visual memory models," in *Proceedings of the 8th ACM international conference on Multimedia*, pp. 95–104, ACM Press, 2000.

50. R. Floyd and L. Steinberg, "An adaptive algorithm for spatial grey scale," in *Journal of the Society for Information Display*, **17(2)**, pp. 75–77, 1976.
51. MoCA: Movie Content Analysis, "Video Color Adaptation for Mobile Devices." <http://www.informatik.uni-mannheim.de/pi4/content/projects/moca/Project-ColorAdaptation.html>, last checked: 19/12/2006, University of Mannheim, 2006.
52. S. Kopf, T. King, F. Lampi, and W. Effelsberg, "Video color adaptation for mobile devices," in *Proceedings of the 14th ACM international conference on Multimedia*, pp. 963–964, ACM Press, October 2006.