Fast Synopsis for Moving Objects Using Compressed Video

Rui Zhong, Ruimin Hu, *Senior Member, IEEE*, Zhongyuan Wang, *Member, IEEE*, and Shizheng Wang, *Member, IEEE*

***Abstract*—With the increasing volume of video data, how to analyze and browse video in a fast and effective way has become an urgent problem in applications. This letter proposes a novel video synopsis method in compressed domain for browsing video captured by static cameras. Synopsis video is a video abstraction, which displays moving objects from different periods simultaneously on the primary background contents of original video. To overcome the low efficiency of traditional video synopsis for compressed video, our method presents a new graph cut algorithm to extract objects tubes and meanwhile gives a fast solution to minimize energy function in compressed domain. Experimental results in H.264 video have demonstrated the high-efficiency of this new video synopsis scheme for massive video browsing.**

***Index Terms*—Compressed domain, graph cuts and video browsing, video synopsis.**

I. Introduction

THE procedures of video synopsis include extraction, tracking and synopsis analysis for moving objects [1]. In moving object extraction, the mainstream methods involve pixel-domain processing. The Gaussian Mixture Model (GMM) based approach [2] takes a blend of multiple Gaussian distributions in temporal domain to model each pixel. For the reason of GMM's high efficiency, it is widely used in moving object extraction and browsing. Furthermore, Heikkil [3] presents a more robust background model based on Local Binary Pattern (LBP) feature. However, high computational complexity would be involved in these pixel-domain methods. To reduce the complexity, Chen *et al.* [4] utilize a Motion Vector (MV) quantization method to preprocess MV and maximum a posteriori estimate to implement moving objects extraction in compressed domain. In [5], Markovian block labeling is used to segment objects based on the classification of MVs. Meanwhile, DCT coefficients can also be applied to extract objects. In compressed domain, [6] and [7] model background with DCT coefficient and MV respectively, which demonstrate the efficiency and effectiveness of compressed-domain background modeling.

A video summarization method is raised based on the analysis of video structures and video highlights in pixel domain [8]. In pixel-domain algorithms, the decoding processing of compressed video leads to high computational complexity. Therefore, a trade-off between performance and efficiency is achieved by an online to offline solution of tubes extraction and synopsis analysis [1]. A compressed-domain extraction and tracking method is proposed to speed up video synopsis [9]. Furthermore, a spatio-temporal graphical model treats block groups as the processing unit to realize moving objects detection and tracking in the compressed domain [10]. Recently, [11] creatively combines object segmentation and tracking together. Comparatively, our contribution focuses on the 3D graph cuts method, through which fast generation of object tubes can be realized by combining objects extraction and tracking.

In paper [1], pixel-domain synopsis analysis, which is the last procedure of video synopsis, is presented and introduced to browse surveillance video in static cameras. Further, scalable synopsis analysis [12] and compressed-domain synopsis analysis [9] are proposed to enhance the practicability of video synopsis. Because that the video to be browsed is usually historical and massive, it will be pretty time-expensive to decode the compressed video to pixel domain [1], [9], [12]. To speed up the video browsing, a more efficient fast video synopsis method is proposed based on compressed-domain graph cut [13], [14].

To this end, objects extraction and tracking are combined by novel 3D graph-cuts model, and the synopsis analysis is implemented in compressed-domain block instead of pixel. Fig. 1 outlines the fast video synopsis scheme, including the synopsis generation in compressed domain and the browsing in pixel domain. Experiments have demonstrated that this efficient scheme facilitates browsing historical video.

The rest of the letter is organized as follows: Section II proposes object tubes extraction method based on 3D graph cuts. Section III discusses the fast synopsis analysis. Section IV shows the experimental results. Finally, conclusions are given in Section V.

II. 3D Graph Cuts Based Fast Tubes Extraction

As shown in Fig. 3, moving object tubes are the 3D spatiotemporal representation of each object taken as the basis of synopsis

Manuscript received February 19, 2014; revised March 20, 2014; accepted April 05, 2014. Date of current version April 25, 2014. This work was supported by Major Program of National Natural Science Foundation of China (NSFC) under Grant 61231015, the National NSFC under Grants 61271256, 61172173, 61172174, and 61303114, and by the the Major Science and Technology Innovation Plan of Hubei Province under Grant 2013AAA020. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ce Zhu.

R. Zhong, R. Hu, and Z. Wang are with the National Engineering Research Center for Multimedia Software and School of Computer, Wuhan University, Wuhan 430079, China (e-mail: zhongrui0824@163.com; hrm1964@163.com; wzy_hope@163.com).

S. Wang is with Chinese Academy of Sciences R&D Center for Internet of Things, Wuxi, China (e-mail: wsz316@foxmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2014.2317754

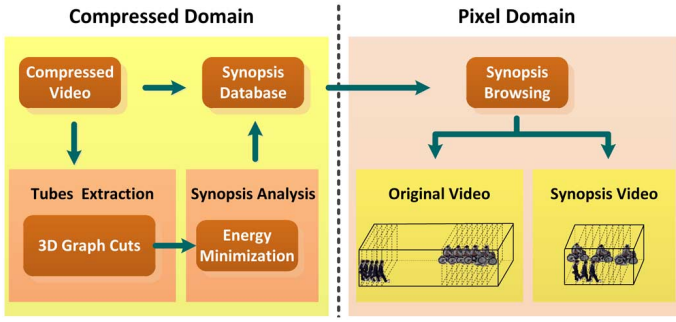

Fig. 1. Flow chart of proposed fast synopsis method: the compressed-domain bitstreams need to be analyzed in compressed domain and decoded into pixel domain to create the synopsis.

analysis. In the case of fast tubes extraction, MV is the main input information for 3D graph cuts. As MV is completely determined by the similarity of pixel values, when a background block is similar to a foreground block, the accuracy of MV is unable to be guaranteed [5]. Hence, in order to ensure MV can reflect the motion trend of moving objects more accurately, MV field should be filtered [4].

In the proposed 3D graph cuts algorithm, the GMM [2] is taken to estimate the initial value of background probability, wherein each 4×4 block as a processing unit. In the initial state, the number of Gaussian components is set as k, the mean μ of the initial value of Gaussian model is $(0, 0)$, and the covariance matrix is $\mathbf{\Sigma} = \sigma^2 I_{2\times 2}$, where $I_{2\times 2}$ is the identity matrix. The weight of the initial Gaussian model is equal to $1/K$. The probability density function that the motion vector $MV^t(x, y)$ at location (x, y) and time t belongs to the k-th Gaussian model is as follows:

$$\widehat{f}_k(MV^t(x,y)) = \frac{1}{2\pi\left|\mathbf{\Sigma}_k^{t-1}\right|^{1/2}} \exp\left(-\frac{(U)^T(\mathbf{\Sigma}_k^{t-1})^{-1}(U)}{2}\right), \tag{1}$$

where, $U = (MV^t(x, y) - \mu_k^{t-1})$. After that, the probability of belonging to background at (x, y) and time t is calculated by: $P^t(x, y) = \sum_{k=1}^{B} W_k^{t-1} \widehat{f}_k(MV^t(x, y))$. Here, B presents the number of background Gaussian distributions and can be calculated through the empirical threshold T_b as below: $B = \arg\min_b (\sum_{k=1}^{b} W_k^{t-1} > T_b)$.

In the initial processing of background probability, the traditional update algorithm of GMM [2] is adopted to update the weight W_k^{t-1}, the mean μ_k^{t-1}, and the covariance matrix $\mathbf{\Sigma}_k^{t-1}$ of Gaussian model at learning ratio 0.5.

After the initial processing of background probability, 3D graph cuts algorithm is further taken to realize the fine extraction of moving objects tubes. The graph cuts item can be presented as the combinatorial optimization process of binary labeling, foreground or background, to each processing unit in graphs. As shown in Fig. 2, the max-flow/min-cut theory in network flow [13] can be utilized to minimize the energy function $E(L)$ for dividing free nodes on the edge of moving objects tubes into

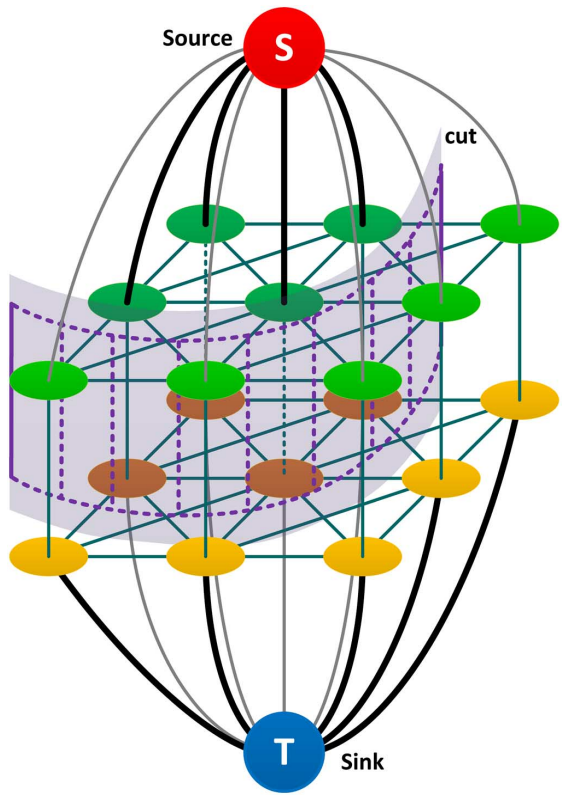

Fig. 2. The principle schematic of 3D graph cut.

foreground or background. Here, each processing unit is treated as a node. In details, we define the energy function $E(L)$ as follows:

$$E(L) = \sum_{p\in P} D_p(L_p) + \sum_{(p,q)\in N} V_{p,q}^{2d}(L_p, L_q) + \sum_{p,r\in T} V_{p,r}^{3d}(L_p, L_r), \tag{2}$$

where $L = \{L_p | p \in P\}$ is the binary label on graph P, $D_p(L_p)$ indicates the penalizing function of differences between the observed value of current node and its true value, $V_{p,q}^{2d}(L_p, L_q)$ represents the penalizing function of differences between spatial neighboring units. $V_{p,r}^{3d}(L_p, L_r)$ signifies the penalizing function of differences between temporal neighboring units pointed by MV.

As shown in Fig. 2, the original graph $I(x, y)$ is mapped to a weighted graph $G(V, E)$ with two terminals, which contains a series of nodes V and weighted edges E connecting nodes. The two terminals of the weighted graph are a source terminal labeling foreground and a sink terminal labeling background. There are three sets of weighted edges in E: N-link, S-link, and T-link. The N-link set contains the edge connecting each unit with the source terminal or sink terminal, and its weight $D_p(L_p)$ guarantees the rationality of labeling unit in Equ. (2). $V_{p,q}^{2d}(L_p, L_q)$ denotes the weight value of the edge in S-link set which connects the spatial neighboring units in the same frame. Meanwhile T-link set corresponds to the edges connecting the temporal neighboring units in two different frames based on MV. The weight of T-link is expressed by $V_{p,r}^{3d}(L_p, L_r)$.

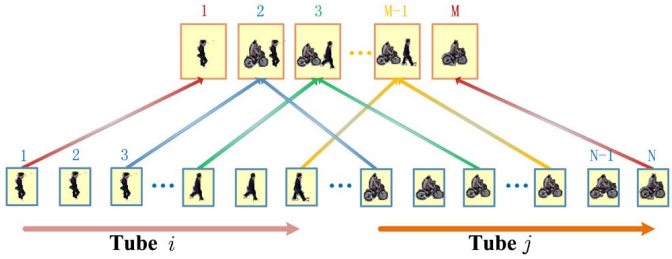

Fig. 3. The mapping diagram of the synopsis video generating.

In the phase of edge weight assignment, according to the definition of bell shaped membership function [15], the probability of a 4×4 block being background is:

$$P_{bg}(P(MV_t)) = \frac{1}{1 + \frac{P(MV_t)}{T_d}}, \tag{3}$$

where T_d is a constant of probability threshold. Thus, for each unit, the probability of being foreground is: $P_{fg}(P(MV_t)) = 1 - P_{bg}(P(MV_t))$.

Consequently, we define the penalizing function $D_p(L_p)$ of difference between the observed value of current node and its true value as the absolute value that foreground probability subtracting background probability:

$$D_p(L_p) = |P_{fg}(P(MV_t)) - P_{bg}(P(MV_t))| \tag{4}$$

As for the penalizing function $V^{2d}_{p,q}(L_p, L_q)$ of difference between spatial neighboring units, we use the reciprocal of Euclidean distance between MVs of neighboring units to represent, the definition is as follows:

$$V^{2d}_{p,q}(Lp, Lq) = \frac{T_s}{Dis^2(p,q) + 1}, \tag{5}$$

where $Dis^2(p,q) = (MV_x(x_p, y_p, t_p) - MV_x(x_q, y_q, t_q))^2 + (MV_y(x_p, y_p, t_p) - MV_y(x_q, y_q, t_q))^2$, and T_s is the regularization parameter, which is proportional to $|MV|^2$ of moving objects. In addition, the penalizing function $V^{3d}_{p,r}(L_p, L_r)$ of difference between temporal connecting units is as below:

$$V^{3d}_{p,r}(L_p, L_r) = \frac{1}{R_{p,r}(MV_t) + 1}, \tag{6}$$

where $R_{p,r}(MV_t)$ is the residual bits number of processing unit after 4×4 transformation [16]. According to above, Equ. (2) can be mapped into a weighted graph with the structure of source and sink terminal [13].

Therefore, as shown in Fig. 2, the max-flow/min-cut theory can minimize the energy function of weighted graph, and stop the optimization when all the free nodes on the edge of moving objects tubes are divided into foreground/background. Intersecting objects can be naturally regarded as an individual object tube and directly acquired by 3D graph cuts.

III. Fast Synopsis Analysis

Video synopsis [1] synopsizes video by rearranging tubes to display objects simultaneously in synopsis video according to the chronological order in original video. The synopsis analysis can be seen as a constrained optimization problem, and the constrains of the solution include the integrity of mapping objects, the chronological order of moving objects, the avoidance of objects collision, and so on.

In the off-line video synopsis, each tube would be labeled with a time tag in the procedure of minimizing energy function $E(\phi)$:

$$E(\phi) = \sum_{i \in Q} E_u(\ell_i) + \sum_{i,j \in Q} E_p(\ell_i, \ell_j) \tag{7}$$

Where Q is the whole tube set, and ℓ_i and ℓ_j signify the time tags of tubes in which the i-th and j-th objects located. ϕ is the mapping relationship of objects between original video and synopsis video, as shown in Fig. 3. E_u and E_p represent the unary and binary energy functions, respectively. The unary energy function E_u is used to penalize the loss of activities in object tubes, and it can be formulated as follows:

$$E_u(l_i) = \sum_{s_r \in b_i} \chi(s_r) - \sum_{\widehat{s_r} \in \widehat{b_i}} \chi(\widehat{s_r}). \tag{8}$$

Where s_r is the r-th block in object b_i of original video, $\widehat{s_r}$ is the r-th block in responding object $\widehat{b_i}$ of synopsis video. $\chi(s_r)$ is the sigmoid area of block s_r in object region. Furthermore, the collision among objects may arise in the synopsis video if it is much short. Thus, we also employ the binary cost function E_p to compute the collision cost at block level:

$$E_p(l_i, l_j) = \sum_{\widehat{s_r} \in \widehat{b_i} \cap \widehat{b_j}} \chi_{\widehat{b_i}}(\widehat{s_r}) \cdot \chi_{\widehat{b_j}}(\widehat{s_r}). \tag{9}$$

To speed up, in compressed-domain synopsis analysis, the smallest computation unit is 4×4 block which results in 1/16 computational complexity of that in pixel domain. Thus, the compressed-domain sigmoid area $\chi(s)$ is used instead of the pixel-domain area $m(s)$ to compute the area of computational unit s. The relationship is:

$$\chi(s) = \frac{1}{1 + e^{-\theta log_2 m(s)}}. \tag{10}$$

θ is the constant used for adjusting the magnitude. Therefore, the synopsis analysis can be formulated into a minimization problem as follows:

$$\phi_{best} = \arg\min_{\phi}(E(\phi)) \tag{11}$$

The minimization of energy function is addressed by Simulated Annealing (SA) [1].

IV. Experimental Results and Analysis

The standard sequences named Hall Monitor [17] and Daytime [12], and F-building [7] are adopted. Experimental environment and parameter setting are as follows: (1) the frame number of synopsis video for these three sequences is 100, as pixel-domain method [12] recommends; (2) the edition number of the core encoder and decoder is JM12.4 of H.264 standard [18]; (3) the entire test sequences are coded in IPPP, and only the first frame is coded as Intra-frame; (4) the quantization parameter (QP) equals to 30, and the MV search range is $[-32, 32]$. All

TABLE I
INITIALIZED PARAMETERS

Sequences	Pixel domain		Compressed domain	
	Init variance (σ)	*Graph-cuts parameter (T_s)*	*Init variance (σ)*	*Graph-cuts parameter (T_s)*
F-building	20.00	—	30.00	5000
Hall	30.00	—	30.00	3100
Daytime	50.00	—	40.00	5000

the experiments are conducted on a desktop PC with Intel Core i5, 2.67 GHz CPU, 2G RAM, and under the Microsoft Windows XP Professional operating system. Initialized parameters of the tubes extraction algorithm are determined empirically, shown in Table I. The proposed algorithm sets the thresholds: $T_b = 0.7$, $T_d = 0.02$ and $k = 3$.

As for the performance evaluation, processing time plays an important role in the browsing of massive video. Therefore, the objective performances are evaluated among all the algorithms in terms of processing time primarily as follows.

For the applications of the browsing in Fig. 1 and the analysis for indexing, our algorithm has achieved respective average 58.79% and 67.10% time-saving against the pixel-domain methods. The main computational complexity is distributed in video decoding and synopsis analysis; and in terms of both, the computational complexity of proposed method is obviously less than pixel-domain methods [1], [12]. Firstly, the information to be decoded in compressed-domain method is only prediction information, such as block partition and MV, rather than residual information for reconstruction in pixel domain. As in Table III, the decoding data volume for synopsis analysis in compressed domain accounts for only 33.5% of that in pixel domain. In addition, as introduced in Section III, since the smallest computation unit in compressed-domain synopsis analysis is 4×4 block, the computational complexity is only 1/16 of that in pixel domain. Besides, objects extraction and tracking are combined by low-complexity algorithms, and thus our method corresponds to lower computational complexity compared with [7], [9].

Furthermore, the three criteria [4], precision (P), recall (R) and F-measure, are used to quantify the objective performance, which are shown in Table II. The 4×4 block-level ground truth is manually labeled. Except for relatively higher recall, both of the precision and the comprehensive performance of [4] are lower than proposed method without pixel-domain refinement. The comprehensive performance of [7] is worse than proposed method because of the lack of graph cuts procedure. Since MV is classified and modeled in a coarse way, the precision and recall of method in [5] are lower than those in proposed algorithm and other anchor methods [4], [7]. In these compressed-domain algorithms, the proposed method has the best objective performance in terms of the F-measure. Due to the limitation of available information in the compressed domain, the performance of moving object extraction in pixel domain is better than compressed-domain methods as shown in Table II.

As for subjective performance, compressed-domain method is good enough for fast browsing as shown in Fig. 4. According to the supported multi-media material, compared with compressed-domain methods [4], [5], [7] without adopting 3D graph cuts, the proposed method improves the subjective effects in terms of temporal-spatial consistence for moving objects by fusing extraction and tracking.

TABLE II
OBJECTIVE PERFORMANCE OF OBJECTS EXTRACTION

Sequence	*Method*	*P*	*R*	*F-measure*
F-building	Proposed	0.55	0.90	0.69
	Pixel-domain	0.76	0.79	0.77
	Ref.[6]	0.46	0.92	0.61
	Ref.[10]	0.54	0.66	0.60
	Ref.[17]	0.59	0.72	0.65
Hall Moniter	Proposed	0.69	0.75	0.72
	Pixel-domain	0.67	0.96	0.79
	Ref.[6]	0.57	0.73	0.64
	Ref.[10]	0.60	0.84	0.70
	Ref.[17]	0.59	0.90	0.71
Daytime	Proposed	0.72	0.66	0.69
	Pixel-domain	0.85	0.89	0.87
	Ref.[6]	0.40	0.86	0.55
	Ref.[10]	0.45	0.60	0.51
	Ref.[17]	0.51	0.87	0.64

TABLE III
DATA VOLUME OF COMPRESSED TESTING SEQUENCES (SIZE: BIT)

Sequences	Pixel domain		Compressed domain	
	Processing Volume	*Rate*	*Processing Volume*	*Rate*
F-building	1,882,400	100%	533,543	28.3%
Hall	1,584,104		605,000	38.2%
Daytime	3,082,080		1,052,116	34.1%

(a) (b)

Fig. 4. The subjective performance in: (a) pixel domain; (b) compressed domain.

V. CONCLUSION

In this letter, a novel method on object tubes extraction and synopsis analysis is presented in the compressed domain. The 3D graph-cuts based synopsis analysis is introduced to extract moving object tubes and obtain the displaying scheme of synopsis video in H.264 compressed domain. As experiments shown, the proposed algorithm works better than the current compressed-domain algorithms in terms of analysis accuracy and outperforms pixel-domain video synopsis in processing time. In the future, we intend to apply our method to the video synopsis for regular video, e.g. movie and TV video.

REFERENCES

[1] Y. Pritch, A. Rav-Acha, and S. Peleg, "Nonchronological video synopsis and indexing," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1971–1984, 2008.

[2] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Comput. Soc. Conf. on CVPR*, 1999, p. 252.

[3] M. Heikkila and M. Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 28, no. 4, pp. 657–662, 2006.

[4] Y. Chen, I. V. Bajic, and P. Saeedi, "Moving region segmentation from compressed video using global motion estimation and markov random fields," *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 421–431, 2011.

[5] W. Zeng, J. Du, and W. Gao *et al.*, "Robust moving object segmentation on h.264/avc compressed video using the block-based MRF model," *Real-Time Imag.*, vol. 11, no. 4, pp. 290–299, 2005.

[6] W. Wang and L. Yang, "Modeling background and segmenting moving objects from compressed video," *IEEE Trans. Ciruits. Syst. Video Technol.*, vol. 18, no. 5, pp. 670–681, 2008.

[7] T. Wang, J. Liang, and X. Wang *et al.*, "Background modeling using local binary patterns of motion vector," *IEEE Vis. Comput., Image Process.*, pp. 1–5, 2012.

[8] C. W. Ngo, Y. F. Ma, and H. J. Zhang, "Video summarization and scene detection by graph modeling[J]. Circuits and systems for video technology," *IEEE Trans. Ciruits. Syst. Video Technol.*, vol. 15, no. 2, pp. 296–305, 2005.

[9] S. Z. Wang, Z. Y. Wang, and R. Hu, "Surveillance video synopsis in the compressed domain for fast video browsing[J]," *J. Vis. Commun. Image Represent.*, vol. 24, no. 8, pp. 1431–1442, 2013.

[10] H. Sabirin and J. Kim, "Moving object detection and tracking using a spatio-temporal graph in H. 264/AVC bitstreams for video surveillance," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 657–668, 2012.

[11] S. Khatoonabadi and I. Bajic, "Video object tracking in the compressed domain using spatio-temporal Markov random fields," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 300–313, 2013.

[12] S. Wang and J. Yang *et al.*, "A surveillance video analysis and storage scheme for scalable synopsis browsing," *ICCV Workshops*, pp. 1947–1954, 2011.

[13] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, 2004.

[14] J. Lu *et al.*, "Moving object extraction algorithm based on graph-cuts in compressed domain," *Comput. Eng. Applicat.*, vol. 12, no. 27, 2013.

[15] I. Gler and E. D. beyli, "Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficients[J]," *J. Neurosci. Meth.*, vol. 148, no. 2, pp. 113–121, 2005.

[16] T. Wiegand, G. J. Sullivan, and G. Bjontegaard *et al.*, "Overview of the H. 264/AVC video coding standard. Circuits and systems for video technology," *IEEE Trans. Ciruits. Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, 2003.

[17] [Online]. Available: http://trace.eas.asu.edu/yuv/hall_monitor/hall_cif.7z

[18] [Online]. Available: http://iphome.hhi.de/uehring/tml/download/oldjm/jm12.4.zip