Video Quality Assurance for SVC in Peer-to-Peer Streaming

Mikko Uitto and Janne Vehkaperä
VTT Technical Research Centre of Finland
Kaitoväylä 1, 90571 Oulu, Finland
mikko.uitto@vtt.fi, janne.vehkapera@vtt.fi

***Abstract*—The Scalable Video Coding (SVC) has begun to arouse interest as a considerable alternative when streaming compressed video over wireless link. The main advantage of SVC comes with the scalability: one encoded sequence can contain multiple decodable sub-streams that allow adaptation to bandwidth fluctuation as well as terminal capabilities. Another growing phenomenon in video streaming is the utilization of peer-to-peer (P2P) technology, which benefits from its reduced costs and load on servers. However, similarly as in non-P2P networks, packet losses and transmission errors are possible, which sets specific need for error resilience especially in the decoder in order to provide sufficient Quality of Experience (QoE). This paper focuses on SVC transmission in error-prone P2P networks and represents our quality assurance strategies focusing mainly to the error concealment for the SVC decoder. The paper also evaluates the effectiveness of the proposed method via simulation setup where the decoder receives incomplete SVC streams that simulate the packet losses in P2P network.**

Keywords-SVC; scalable video coding; error concealment; P2P; peer-to-peer; QoE

I. INTRODUCTION

Video streaming due the fast development of Internet and video technology demands new ways of streaming high quality video to users with different network and terminal device capabilities. Indeed, in the side of need for better transmission technologies the increasing number of different end terminals has awaken the need for dedicated video compression technologies. One of the strong candidates is Scalable Video Coding (SVC), which requires only one encoding for the video, but multiple sub-streams can be decoded from the single stream. This allows not only considerable bandwidth adaptation but also excellent suitability into the receiving terminal. This means that the same video can be streamed both to high quality television with extremely high quality as well as low resolution mobile phones with lower quality.

Alongside the development of video compression, better ways for streaming real-time video are needed that reduce the load of the dedicated video servers [1]. Peer-to-peer (P2P) technology has aroused interest as an alternative transmission gateway when streaming video content among several users. One of the advantages comes with the non-dedicated server implementation since each user works as a server as well as a client. Peer-to-peer structure also reduces maintenance costs and provides simplicity although more work needs to be done in future with the security issues [2]. Content Delivery Network (CDN) support often large number of users likewise in P2P, but they require the deployment of special infrastructure [3].

Despite the fast development of powerful video codecs and streaming techniques a chance for transmission errors is always present due e.g., to network congestion, delay requirements and high video bitrate [4]. Additionally to these issues, temporary link failure can cause significant QoE degradation or even crash the decoder. Some of the research done for SVC quality assurance in P2P relies on controlling the errors via Forward Error Correction (FEC), Flow Forwarding (FF) or retransmission [1] without considering the loss potential still in the decoder. Available tools for maintaining the SVC video quality for spatially oriented streams via error concealment have been investigated [5] and also implemented to the old SVC reference software [6],[7]. However, these are mainly developed for individual frame losses and therefore require, for example, complex memory usage in order to maintain the previous pictures as a reference to the following pictures. Additionally, complex data structures in the error concealment can set hard limitation for the real-time performance, also for low-resolution streams. Furthermore, none of the existing quality assurance and error concealment techniques are designed especially to P2P streaming.

In this paper, we describe some of the work done in P2P-Next project, which is a research project funded partially by European Commission in the context of Framework Program 7 [8]. One of the goals in this project is to develop a P2P content delivery platform, the NextShare system with SVC support. Without going into deep in SVC integration into P2P architecture of the NextShare system we focus in this paper how to maintain the satisfying quality among the end users when packet losses are possible during the SVC transmission concentrating especially on the error concealment possibilities in the decoder.

The paper is organised in the following way. In Section 2, we describe SVC delivery in P2P architecture and present the NextShare platform. In Section 3, we provide our approach how to maintain video quality on a satisfying level for the end user. In Section 4, we evaluate our approach and compare it to the reference cases via simulation setup. Finally, Section 5 concludes the paper.

II. SVC IN PEER-TO-PEER ARCHITECTURE

The cost-effective solution of P2P has aroused widely interest as an alternative gateway for real-time video streaming. However, the number of such systems with full SVC support [9] is rare although the recent research has investigated this to some extent [10]. The majority of the research in this area, such as LayerP2P [11] does not consider SVC as the applied video codec but rely on non-layered codecs, such as H.264/AVC.

A. SVC advantages

The MPEG-4 Scalable Video Coding standard is an extension of the H.264/AVC standard (AnnexG) and provides a number of different layers within one encoded bitstream. While the H.264/AVC compliant base layer of a scalable bitstream provides the minimum quality, the enhancement layers are used to further increase the quality, resolution or frame rate of the bitstream [12]. Thus, a client only needs to receive a small part of the scalable bitstream to consume the data in low quality, while it has to receive and decode the complete scalable bitstream to consume the data in best quality. The usage of scalable codecs simplifies the adaptation of bitstream significantly, as an adaptation of such a bitstream can be performed by simply skipping some or all of the data related with enhancement layers.

The SVC base layer may be enhanced in three dimensions: the temporal dimension (frame rate), the spatial dimension (resolution) and the quality dimension (SNR) [12]. When considering networks with fluctuating bandwidth, especially temporal and SNR scalabilities enable powerful adaptation by diminishing the video bitrate. However, when several terminals with unique device capabilities exist also spatial scalability is a considerable alternative for saving the encoding time of various different types of sequences.

An essential feature of the design of the SVC extension is that the majority of the components of the H.264/AVC standard were adopted. This implies that transform coding, entropy coding, motion compensation, intra-prediction, the deblocking filter or the structure of the NAL units (NALU) are used as intended for the H.264/AVC standard [12]. One advantage of this approach is that the base layer of an SVC-encoded bitstream can generally be processed by a H.264/AVC compatible decoder, as the extensions of the H.264/AVC standard are only used to support spatial and signal-to-noise ratio (SNR) scalability.

B. NextShare

The NextShare, an open-source system, the next generation P2P content delivery platform, is developed in the P2P-Next project [8] and it has a fully support also to SVC. Basically, it follows the foundation of BitTorrent, but thanks to the NextShare development of state-of-the-art scientists, it can be now used not only for single layer streams, but also to multi-layered SVC streams. The basic principle and also a benefit in this platform is that the core won't require any changes if the video codec e.g., the decoder needs to be replaced to another. Additionally, the following error concealment as well as quality assurance technique presented in the next section is so called stand-alone algorithm that is not decoder dependent. This means that its integration is done basically to the decoder-player interface without requiring any major changes to the decoding process.

The simplified overall model of the producer-consumer side architecture can be seen in Fig. 1. The current SVC implementation in NextShare is designed to support both spatial and SNR layers. Likewise in the evaluation section of this paper, we have modelled the system with 4-layer mixed scalabilities where both the base layer BL and the spatial enhancement have one additional SNR layer (see TABLE 2). In the NextShare setup [9], we use 64 frames in a piece with 25 fps, with 3072 Kb/s for the highest VGA high-quality layer. Naturally, all the layers are mapped to pieces separately. The SVC encoder is optimised to have a constant bit rate with only one slice in a picture, because the coding efficiency suffers from using multiple slices [13].

Furthermore, the P2P engine that is responsible for creating and injecting the content into the network will not send the upper layer before the corresponding lower layer is sent [9]. In addition, the pieces are sent forward only if all the frames are received, which means that individual frame losses are not possible. Since the decoder will receive only "complete" group of pictures (GOPs) it guarantees in theory that the decoder should never crash. However, problems arise especially when spatially scalable video is streamed. First, the user may experience that the resolution varies in the player, which can be a very annoying phenomenon. Second, it is not always certain that the decoder is able to survive from the layer switching, especially if no Intra Decoding Refresh (IDR) pictures are used. This means that error concealment is needed to assure the video quality.

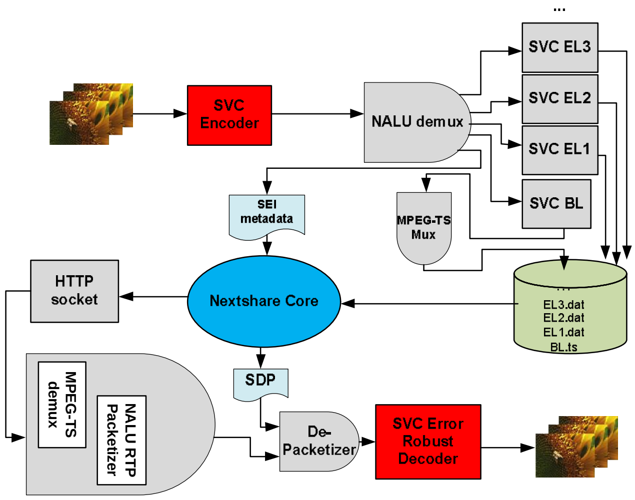

Figure 1. Simplified model of the producer-consumer side architecture.

III. PROPOSED QUALITY ASSURANCE TECHNIQUES

The available bandwidth does not always guarantee lossless transmission of the video stream. Especially in P2P networks the number of peers can vary causing total enhancement layer GOP losses in the receiving terminal. This means that the decoder must take these losses into an account and provide not only a stable and steady decoding process but also a satisfying video quality for the end users.

The error concealment in H.264/SVC decoder is to ensure complete decoding without crashes and to provide sufficient quality of experience. In some cases the spatial enhancement layer(s) cannot be received within the defined time slots, especially with high data rate videos. One problem in SVC decoder is that the first IDR packet usually defines the target resolution to be decoded: if the highest layer of spatially scalable stream cannot be received, it can crash the decoder or the resolution may vary from high to low, which can be very annoying phenomenon for human eye. Another viewpoint is that the hierarchical prediction structures in SVC can cause extensive error propagation.

We took the SVC reference decoder (version 9.15) [6] as a starting point and implemented error concealment for the decoder in order to provide good error robustness. Second, we concentrated on implementing picture upsampling techniques because the varying resolution in the player, such as in VLC, is a very provocative quality of experience.

Currently, JSVM reference software provides four separate upsampling algorithms with the picture resampling tool [14]. The first upsampling method is based on integer-based 4-tap filters that are originally derived from Lanczos-3 filter and arbitrary upsampling ratios are supported. On the contrary, the second method supports only dyadic upsampling ratios where the actual upsampling process is performed with several dyadic stages using also interpolation for the missing luminance and chrominance samples. The third method applies three-lobed Lanczos-windowed sinc functions and finally, the fourth method is the combination of AVC-half sample interpolation filter and bi-linear filter. [14]

We implemented the first method, integer-based 4-tap upsampling filter, as a separate function after the decoding process in order to enable an easy integration into different decoders. We optimised the time consuming blocks via benchmarking and achieved real-time algorithm, at least for VGA resolution video.

The use of IDR pictures is one easy way to break the decoding chain and check whether all the spatial layers are received. Therefore, we monitor the resolution of the first IDR picture, taking place as a first picture in each GOP. As was presented in Section II, the P2P engine sends only full GOPs to the consumer. So we basically upsample the whole GOP until the next IDR is received. Fig. 2 clarifies the upsampling process.

The actual upsampling process is simple; the algorithm solely takes the decoded picture as an input and upsamples it into the target resolution, which is defined in the Sequence Parameter Set (SPS) NAL packet. After this, the upsampled picture is directed either in the file writing process or to video output player, such as VLC. On this work, we focused principally implementing "portable" upsampling routine that guarantees satisfying end quality. The next step will be to apply and/or develop even better filters, which are state-of-the-art [15]. However, in this work we also benchmarked the JSVM filters and implemented the one providing the best end quality. This can be seen TABLE 1, which illustrates the sequence average Peak-Signal-to-Noise-Ratio (PSNR) comparison results in decibels (dB). Clearly our choice, the 4-tap filter, provides the best end quality both when upsampling the spatial base layer (BL) or its first quality enhancement layer (EL1) where Coarse-Grain Scalability (CGS) is used. The three test sequences will be introduced later in Section IV.

TABLE 1. JSVM upsampling filter comparison.

	PARKRUN		*SUNFLOWER*		*CREW*	
	BL	EL1	BL	EL1	BL	EL1
4-tap	21,80	23,04	28,46	33,02	29,48	32,50
Dyadic	21,43	22,39	27,51	30,27	29,20	31,68
Lanczos	21,43	22,37	27,52	30,28	29,20	31,66
Half-pel +bilinear	21,43	22,39	27,51	30,27	29,20	31,68

We did not want to focus only on simple spatial scalability when outlining the upsampling implementation. Instead of this, we used mixed spatial and quality layer scenario that was defined already in the project [9]. This enables a configuration where multiple receiving terminals with different device capabilities exist.

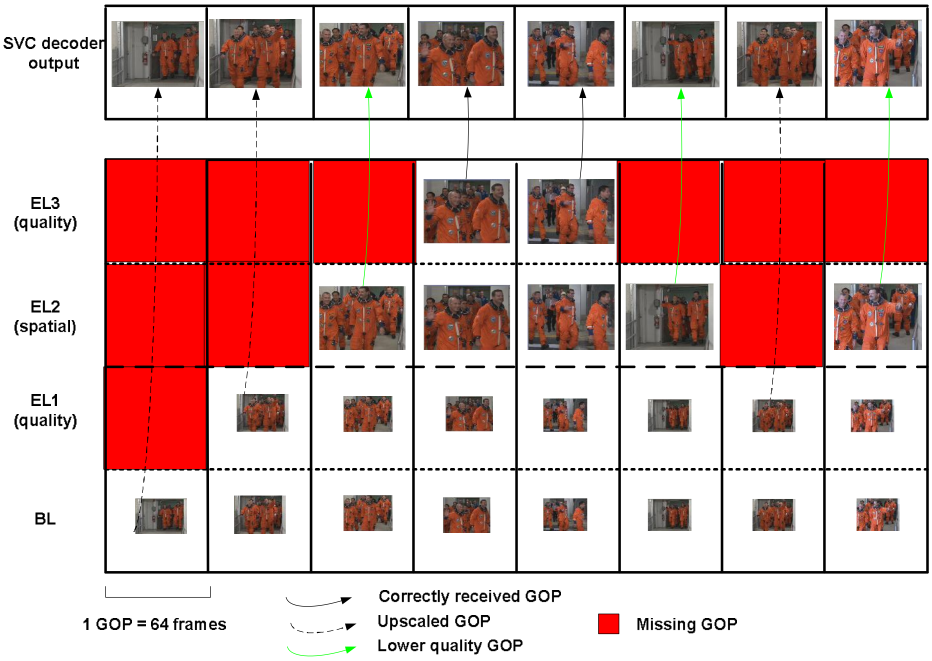

Figure 2. SVC upsampling process.

IV. EVALUATION

The evaluation section consists of the description of the simulations and their results. In addition, the corresponding SVC decoder is also integrated into the SVC prototype and the functionality of the error concealment is confirmed with the actual P2P setup as well [9].

A. Simulation setup

In order to evaluate the effectiveness of our quality assurance technique for the SVC decoder we developed also a packet loss generator as separate software that reads the SVC stream and drops packets with a certain loss ratio. For the purpose of P2P packet loss simulations we modified the software to drop whole GOPs from the stream. Basically the GOPs were dropped randomly but in a manner where the missing GOP and its higher enhancements were also discarded. Consequently, we were able to replicate comparable model for the P2P video decoder in the aspect of transmission errors. Once generating the GOP losses we decoded the output file with our modified SVC decoder and then measured the output PSNR. We repeated this simulation chain 50 times for each GOP loss ratio (2%, 5%, 10% and 20%) in order to average the PSNR values for each frame.

We chose three sequences with different characteristics mainly to have variety in the results (see Fig. 3). The *Parkrun* illustrates a running person both with steady slow motion, moving camera as well as static scenes with zero motion. The second sequence, *Sunflower*, contains only sharp motion in a small area both from the camera and the bee. The final sequence, *Crew*, contains lot of motion, bright lights and colors.

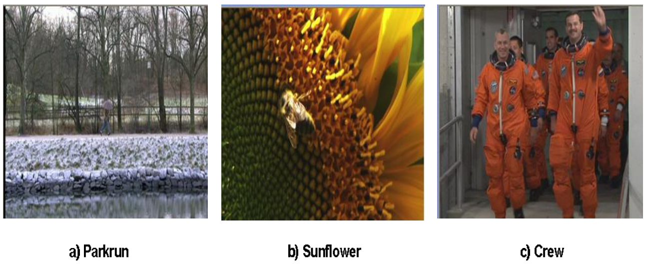
a) Parkrun
b) Sunflower
c) Crew

Figure 3. Test sequences.

The encoding parameters can be seen in TABLE 2. In order to have a variation to PSNRs and bitrates we encoded the test sequences without enabling the rate control. However, in the real demonstrator constant bit rate is applied but for our simulations for the error concealment it is not needed. We used CGS for quality enhancement laeyrs. As was illustrated in Fig. 2, we upsample the highest received GOP. If the GOP loss generator drops only the highest layer, then the output GOP is decoded with the quality from EL2. Furthermore, if EL1 is the highest received layer, we decode this one and use it as a reference for the upsampling algorithm.

TABLE 3 presents the encoded PSNRs for each layer. Naturally, when e.g., upsampling the EL1 (*Parkrun*) the end PSNR is not anymore 33.65 decibels for the high-resolution image, because upsampling causes blurriness to some extent.

TABLE 2. SVC encoding parameters.

Number of layers	4
BL & EL1 resolution	QVGA (320x240)
EL2 & EL3 resolution	VGA (640x480)
GOP size (IDR period)	64
B frames	yes
Frame rate	30 fps
BL bitrate (*Parkrun*;*Sunflower*;*Crew*)	100; 70; 200 kb/s
EL1 bitrate (*Parkrun*;*Sunflower*;*Crew*)	700; 600; 800 kb/s
EL2 bitrate (*Parkrun*;*Sunflower*;*Crew*)	1300; 800; 900 kb/s
EL3 (full) bitrate (*Parkrun*;*Sunflower*;*Crew*)	4500; 2000; 2400 kb/s

TABLE 3. Encoded PSNRs for each layer.

	Parkrun	*Sunflower*	*Crew*
BL	27,96	29,55	30,80
EL1	33,65	35,61	35,50
EL2	26,87	32,51	32,49
EL3	32,96	37,31	36,08

B. *Results*

This section presents the results of the simulations where random GOP losses were injected to the three 1800-frame sequences. Fig. 4 – Fig. 6 show the PSNR-Y curves as a function of GOP loss ratio %. As can be seen in all simulation cases our proposed SVC quality assurance as well as error concealment technique outperforms the reference case, which was the so called “frame freeze” technique that can be widely seen in various video players. We can observe that the PSNR difference between the proposed and the reference case is only 2 dB (for 20% loss) for the *Parkrun*. This can be explained by the sequence characteristics where basically the video background and main target remain the same all the time creating smaller gaps between BL and EL3 PSNRs.

For the other two test sequences, the PSNR variation at 20% ratio is approximately 3-4 dBs better and it is clearly seen that the PSNR difference would increase for greater GOP loss ratios. Despite the fact that the end quality is significantly better as the PSNR values indicate, the visual quality, especially jerkiness, is extremely smooth without any freeze states in the video playback. As can be seen for the *Crew* sequence in Fig. 7 the overall quality improvement with the proposed method is significant.

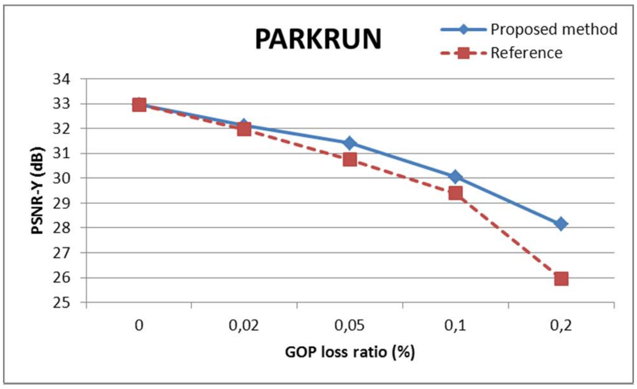

Figure 4. Average PSNR results for the *Parkrun* sequence.

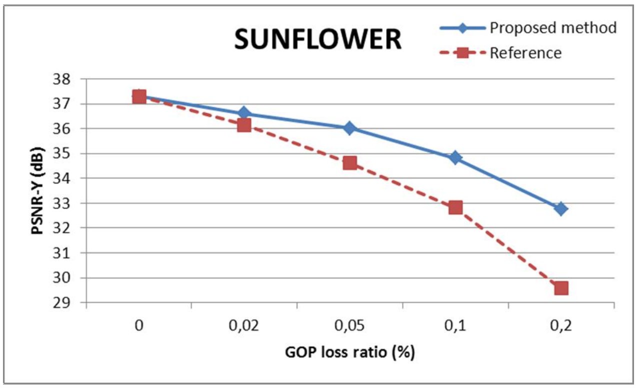

Figure 5. Average PSNR results for the *Sunflower* sequence.

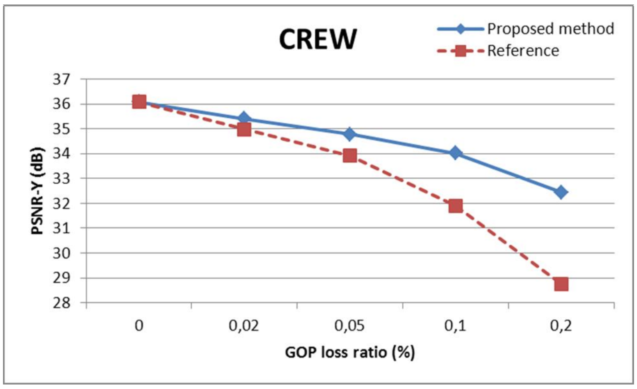

Figure 6. Average PSNR results for the *Crew* sequence.

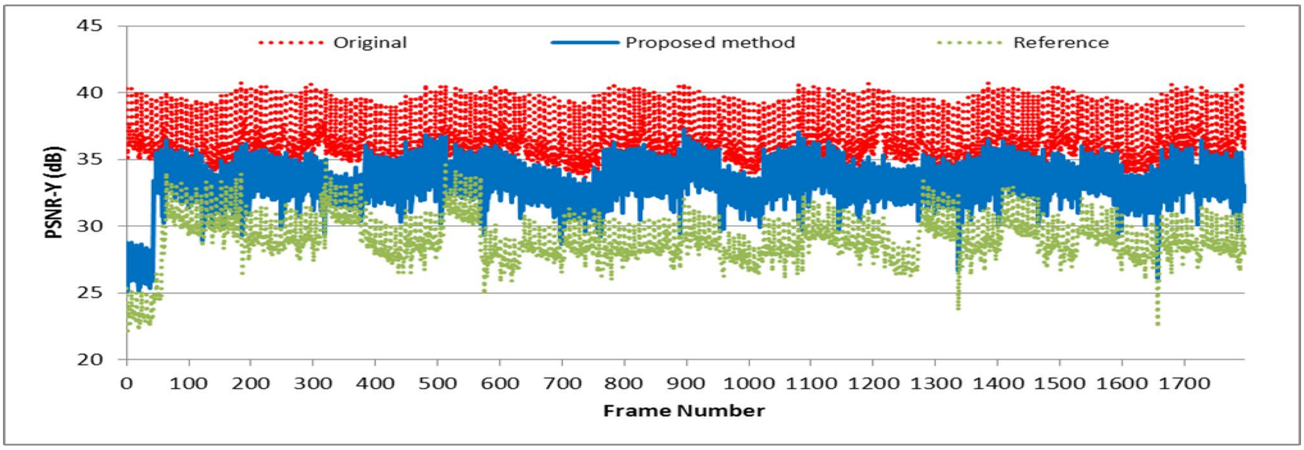

Figure 7. PSNR curve for Crew – sequence (1800 frames).

V. CONCLUSIONS

This paper investigated how to maintain the quality of experience in a good level for SVC streams in P2P streaming. The paper introduced the actual P2P platform developed in the P2P-Next project and its SVC relevance. We presented our implemented technique for video quality assurance focusing on the SVC decoder-side error concealment possibilities. In addition to the fact that the upsampling implementation is also running in the Nextshare SVC platform, we made our own simulation setup in order to evaluate the goodness of the quality assurance technique. The results show inevitably that our approach provides a lot smoother visual quality of experience compared to the traditional frame-freeze technique and also the computational values via PSNR curves proves that our method is applicable algorithm to be used in the SVC decoder. The proposed algorithm will be a portable block between any video decoder and player in future.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union's Seventh Framework Programme (P2P-Next) under grant agreement n° 216217. The authors would like to thank for the support.

REFERENCES

[1] T. Schierl, Y. Sanchez, C. Hellge, and T. Wiegand, “Improving P2P Live-Content Delivery using SVC”, Proceedings of IEEE Visual Communications and Image Processing (VCIP'10), Huang Shan, An Hui, China, pp. 1-10, July 2010.

[2] P. Sanjoy, “Digital Video Distribution in Broadband, Television, Mobile, and Converged Networks: Trends, Challenges and Solutions”. John Wiley & Sons Ltd, West Sussex, United Kingdom, pp. 1-384, 2011.

[3] P. Bacciechet, T. Schierl, T. Wiegand, and B. Girod, “Low-delay Peer-to-Peer Streaming using Scalable Video Coding”, Packet Video 2007, Nov. 2007, pp. 173-181, doi: 10.1109/PACKET.2007.4397039

[4] B. Li and H. Yin, “Peer-to-Peer Live Video Streaming on the Internet: Issues, Existing Approaches and Challenges”, IEEE Communications Magazine, Vol. 45, June 2007, pp. 94-99, doi: 10.1109/MCOM.2007.374425

[5] Y. Guo, Y. Chen, Y.-K. Wang, H. Li, M. M. Hannuksela, and M. Gabbouj, “Error Resilient Coding and Error Concealment in Scalable Video Coding”, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 19, pp. 781-795, Jun. 2009, doi: 10.1109/TCSVT.2009.2017311

[6] SVC Reference Software (JSVM software). [Online]. Available:

http://ip.hhi.de/imagecom_G1/savce/downloads/SVC-Reference-Software.htm, accessed on 23/06/2011

[7] Y. Chen, K. Xie, F. Zhang, P. Bandit, and J.Boyce,” Frame loss error concealment for SVC”, Journal of Zhejiang University – Science A 7, pp. 677-683, December 2006.

[8] The P2P-Next Project, FP7-ICT-216217, http://www.p2pnext.org, accessed on 23/06/2011

[9] N. Capovilla, M. Eberhard, S. Mignanti, R. Petrocco, and J. Vehkaperä, “An Architecture for Distributing Scalable Content over Peer-to-Peer Networks”, Proc. Advances in Multimedia (MMEDIA), 2010, pp.1-6, June 2010, doi: 10.1109/MMEDIA.2010.17

[10] M. Mushtaq and T. Ahmed, “Smooth Video Delivery for SVC based Media Streaming over P2P Networks”, Proc. Consumer Communications and Networking Conference 2008 CCNC 2008 5th IEEE, pp. 447-451, Jan.2008.

[11] Z. Liu, Y. Shen, K. W. Ross, S. S. Panwar, and Y. Wang, “LayerP2P: Using Layered Video Chunks in P2P Live Streaming”, IEEE Transactions on Multimedia, Vol. 11, pp. 1340 – 1352, August 2009.

[12] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard", IEEE Trans. on CSVT, vol. 17, no. 9, pp. 1103-1120, September 2007.

[13] M. Uitto, J. Vehkaperä, and P. Amon, ” Impact of Slice Size for SVC Coding Efficiency and Error Resilience”, 6th International Mobile Multimedia Communications Conference (MobiMedia 2010, SVCVision Workshop). Lisbon, Portugal, pp. 1-15, 6-8th September 2010.

[14] JSVM 9.15 Software Manual, 2008.

[15] Q. Shan, Z. Li, J. Jia, and C.-K. Tang, “Fast Image/Video Upsampling”, ACM Transactions on Graphics (SIGGRAPH ASIA 2008), Vol. 27, pp. 1-8, November 2008.