Chapter 6
Conclusions and Future Works

To serve multimedia applications over the mobile environment, this dissertation thoroughly discussed several issues including cost, power and channel impairment. In Chapter 2, we efficiently integrate MPEG-2 and H.264/AVC for multi-standard requirements. In Chapter 3, we have pointed out that current approach is still not applicable to the power requirement for portable mobile devices. To alleviate this problem, we propose several low-power techniques to reduce memory cost and transmission bandwidth. Moreover, Chapter 4 presents a novel error detection and concealment for a robust transmission of video data. Overall, implementation and measured results exhibit that 303.78KGates/22.75Kb H.264/AVC and MPEG-2 video decoding in QCIF@15fps are achieved at 1.15MHz clock frequency with power dissipation of 125μW and 108μW respectively at 1-V supply voltage. This low-power and area-efficient feature makes our proposal very suitable for mobile applications. In the following, three conclusions have been made based on the aforementioned design aspects. They are low-cost, low-power and error-robustness which are related to the design challenges in a typical mobile system. After that, we also point out several design topics for further researches.

6.1 Conclusions

6.1.1 Dual-Mode Video Decoder for Multi-Standard Requirements

A single-chip MPEG-2/H.264 decoder is developed which cut down the cost of video decoding side. The cost reduction is attained mainly by integration of IDCT, deblocking filter, and entropy decoder. We adopt many design techniques both on algorithmic and architectural levels. In particular, we exploit a recursive algorithm and resource sharing technique to improve the hardware utilization in IDCT module. The proposed deblocking filter not only supports H.264/AVC specification but also improves the visual quality in MPEG-2 video decoding. Moreover, several VLC tables have been merged to reduce the table size in entropy decoding. Overall, in a 0.18μm CMOS process, a dual-mode video decoder is implemented for multi-standard requirements and achieves 303.78K of logic gates which is approximately 80% of cost as compared with a separate video decoder.

6.1.2 Low-Power Implementation for Portable Devices

A sub-mW H.264/MPEG-2 video decoder for portable devices is realized, where three innovative low-power techniques have been proposed. Specifically, the DPS method is used extensively to optimize the register numbers and processing cycles. Three-level memory hierarchy and LPL scheme achieve better memory allocation and access efficiency, respectively. Low-power motion compensation and deblocking filter are designed to lower the required working frequency. Additionally, supply voltage scaling is further applied in order to reduce the power dissipation by lowering V_{DD}. Totally the power reduction of this design is about one order of magnitude compared to state-of-the-art implementations. To obtain a real metric of power, a test chip is fabricated in a 0.18μm 1P6M CMOS technology. For mobile applications, H.264/AVC and MPEG-2 video decoding of quarter-common intermediate format (QCIF) sequences at 15 frames per second are achieved at 1.15MHz with power dissipation of 125μW and 108μW respectively at 1-V supply voltage.

6.1.3 Improved Visual Quality over Mobile Environments

A quality-improved MPEG-2/H.264 video decoder is presented which is robust to channel disturbance by early detecting and thereby concealing corrupted regions in a frame. A joint design of source and channel decoding is achieved by receiving soft-information of de-modulated symbol. This information directly feeds through video decoder for enhancing error-robustness. It not only provides the reliability of each decoded bit but also benefits error detection and concealment modules in the video decoding side. Simulation results reveal that more than 1dB of PSNR can be gained under 2.7×10^{-3} of BERs. Therefore, this proposal is more robust to channel behaviors and applicable to mobile communication environments.

6.2 Future Works

6.2.1 Frame Re-compression Algorithm

While a literature review of frame re-compression methods had been discussed in Chapter 4.5.1, there are many design challenges left to be resolved. An ideal frame re-compression features high compression ratio, simplicity, low compressing latency, and random accessibility. To achieve different dimensions of design features, many solutions have been proposed but cannot meet all features in a single algorithm. Moreover, based on algorithms so far proposed, both simplicity and random accessibility are the upmost factor to achieving success of frame re-compression. In the simplicity point of view, Lee [84] commented that high compression ratio sacrifices the feature of design simplicity. In [84], a simple compression algorithm is realized but a compression ratio over 50% is not recommended because simplicity, low latency, and random accessibility are more important in real-time decoding requirements. However, there is doubt whether this recommendation is definitely true for any compression algorithms. One could further investigate a new algorithm to combat this recommendation or improve the compression ratio without sacrificing other design features. On the other hand, Bourge and Jung [100] pointed out that the main drawback of compression algorithms is the random accessibility. They named this drawback as data overhead which is defined as the ratio between the number of pixels that are actually accessed during the motion compensation (MC) of a block and the number of pixels that are really useful in the reference block. This ratio is superior to one because of the block-based nature of compression techniques. For instance, let's consider that the reference frames are encoded on an 8×8 block grid. If MC wants to access the value of one pixel within 8×8 block we must fetch and decode the whole block it belongs to. In other words, the compression gain will degrade due to the additional accesses for reading one pixel data and that's why the data overhead is greater than 1. This phenomenon is thoroughly discussed in [100] but do not give a complete solution to overcome the problem of random accessibility.

When the goal of frame re-compression is purely to reduce the required memory capacity for cost reasons, virtually any efficient still image compression techniques can be performed. In practice, we pay more attention on lossless image compressions [117][118] because lossy compression comes at the expense of a quality degradation that worsens along a group of pictures. Although those compression algorithms intend to compress still image, they can be considered as a frame re-compression method after a slight modification. Therefore, it becomes an applicable candidate for further researches in the frame re-compression algorithms.

6.2.2 Joint Source and Channel Design (JSCD)

In the past, the designs of source and channel coder have been performed separately. This often makes excellent senses and could be proved by the separation theorem of Shannon [119]. However, Shannon's theorem effectively assumes that source coder removes all data redundancy, and the channel coder inserts additional redundancy to protect the source data due to the impairment of physical channel. This separation does not make as much practical senses. It has been shown that the separation theorem does not hold for all channel conditions [120]. To achieve better performance between source and channel coding sides, the idea of joint source and channel design (JSCD) has been gaining increasing attention in recent years due to the significant growth of multimedia wireless communication over the noisy and band-limited channels. Actually, soft CAVLC decoder in Chapter 4.3 can be considered as a case study in this design field. The CAVLC source decoder involves soft information from channel coding side to improve the error robustness. However, the complexity and memory storage issues are still of great challenge [71]. Moreover, this joint design retrieves the soft stream to be aware of signal reliability. The reliability problem also impacts the decoding procedures in the source decoding side. For example, the proposed soft CAVLC decoder improves the detection capabilities of erroneous pixels in macroblock levels, but how to determine whether the detected results are correct or not is the reliability issue. This issue should be resolved and thereby considered in the follow-up source decoding processes. In addition to complexity, storage, and reliability issues, integrating soft designs into the whole coding system over error-prone environments would be a tough task. The reason is that existing video and communication standards are defined without taking JSCD concept into accounts. For instance, H.264/AVC video standard defines data partition modes in extended profiles. The insignificant part includes not only coefficients for CAVLC decoder but also other header information such as MV, CBP, etc. The additional inclusion of header information will make the soft CAVLC decoder unpractical in the existing video standards. Moreover, the existing communication standards, DVB-H, do not expect to provide soft information for source decoder. Hence, a slight modification is required for simulating the JSCD concepts over the existing systems, and therefore the standard-incompatible problem will be emerged.

6.2.3 Scalable Video Coding (SVC)

Scalable video coding has been an active research area for about 20 years. The prevalent video standards MPEG-2 video [2] and MPEG-4 visual [3] already include several tools for supporting temporal, spatial, and SNR scalabilities. However, the scalable profiles of these standards have been rarely used, mainly because those scalabilities come along with a significant loss in coding efficiency as well as an increase in decoder complexity. To improve the coding performance, the Joint Video Team of the ISO/IEC Moving Picture Experts Group and the ITU-T Video Coding Experts Group is currently standardizing a scalable extension of their video coding standard H.264 / MPEG-4 AVC. This SVC project has achieved a significant improvement in coding efficiency and the supportable degree of scalability relative to the scalable profiles of existing video coding standards. However, the potential increase of computational complexity still becomes a big problem for future acceptance in the industry. As we will see in an overview manuscript of SVC standard [121], hierarchal B frame, inter-layer prediction and fine grain SNR scalability (FGS) have been adopted for temporal, spatial and SNR scalability, respectively. Those coding tools will definitely increase the computational complexity in the source coding/decoding systems. On the other hand, the newly-announced SVC standard involves several streaming, networking, and communication design concepts [122]–[125], leading to the higher barrier for the knowledge of SVC algorithm. Hence, it is believed that the source and channel coding have to be jointly considered for next-generation video designs.

6.2.4 Multi-view Video Coding (MVC)

While the H.264/AVC standard is beginning to see adoption, a number of activities are already extending the core H.264/AVC standard. One key extension to H.264/AVC currently undergoing standardization is Multi-view Video Coding (MVC) [126]. The MVC standardization activity began in early 2006 and is expected to be available in early 2008. It specifies tools for jointly compressing the multiple views of the same scene captured using multiple cameras. The MVC video is applicable to Free Viewpoint TV (FTV) and 3D-TV. Figure 6.1(a) shows a FTV system [127] which can generate very natural free viewpoint images in real time as shown in Figure 6.1(b). A scene is captured using a dense array of synchronized cameras. The camera images are placed in a simple manner that allows rendering the scene from any position by simple processing. So the user can view the scene not only from the original camera positions but also from any virtual viewpoint. FTV supports a wide variety of applications since it can be applied to any kind of scene. On the other hand, Figure 6.2 shows an example of a 3D-TV system. A scene is again captured by *N* synchronized cameras [128]. The multiple video signals are encoded and transmitted. At the receiver they are decoded, rendered and displayed on a 3D display. 3D rendering means creating 2 views, one for each eye, which if perceived by a human will create a depth impression. There are several types of 3D displays, with and without glasses. Overall, to clarify the 3D-TV from a visual perspective, an on-line demo is available in [129]. This is the first trial to accomplish a 3D-TV system that allows for real-time acquisition, transmission, and 3D display and is expected to be the next revolution in the history of TV.

(a) (b)

Figure 6.1: (a) Capturing and processing parts of FTV system. (b) Free viewpoint images generated in real time.

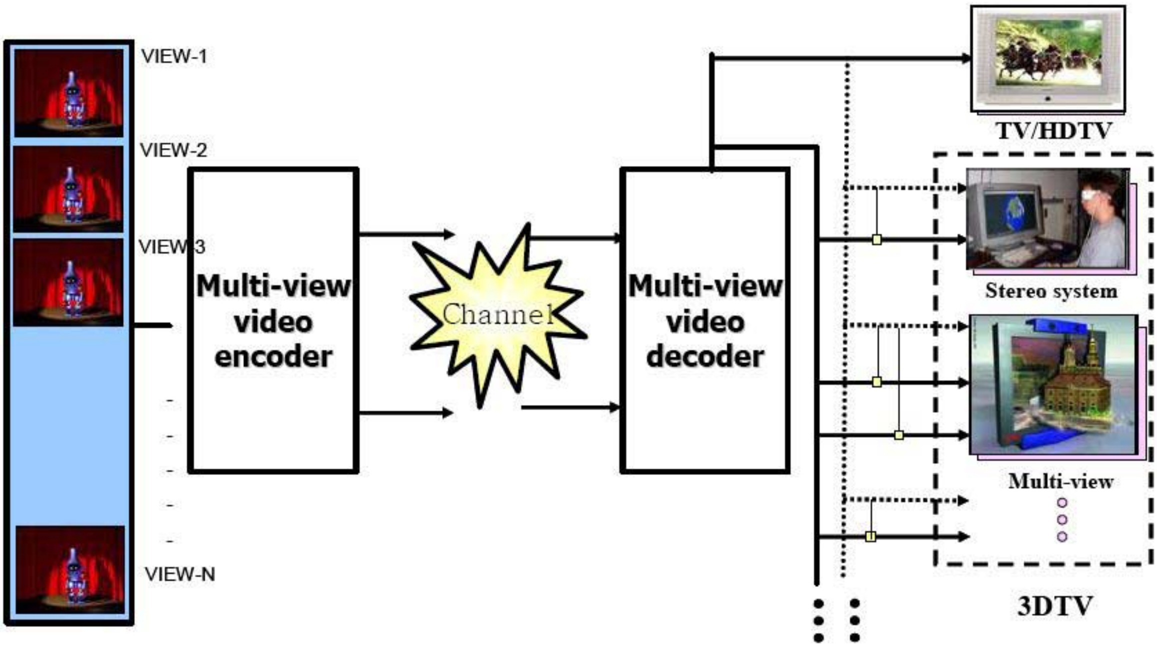

Figure 6.2: Example of a 3D-TV system.

A common element of aforementioned systems is the use of multiple views of the same scene that have to be transmitted to the user. The straight-forward solution for this would be to encode all the video signals independently using a state-of-the-art video codec such as H.264/AVC. However, Muller *et al.* [130] has been shown that specific multi-view video coding (MVC) algorithms give significantly better results compared to the simple H.264/AVC simulcast solution. The basic idea is to exploit spatial and temporal redundancy for compression. Since all cameras capture the same scene from different viewpoints, spatial redundancy can be expected. Moreover, images are not only predicted from temporally preceding images but also from corresponding images in adjacent views. Besides such spatio-temporal prediction structures, specific prediction tools have been proposed that can be combined with any prediction structure. This includes for instance illumination compensation, spatio-temporal direct mode, disparity/motion vector prediction, and view interpolation [133]. Among those prediction tools, view interpolation prediction and illumination compensation seem to be very promising for further improvement. The latter sub-section gives a brief explanation of these modules without detailing the design algorithms or features.

6.2.4.1 View Interpolation Prediction

When viewing a stereo pair on a flat screen, the right and left eye converge as if portions of the image are located at different distances, yet they continue to focus on the plane of the screen. This breakdown in the normal accommodation/convergence relationships is a known cause of eyestrain, and the severity of the strain is related to the disparity of the stereo images [131]. The view interpolation method [132] can be used to overcome this problem and can also be used for multi-view coding. In general, view interpolation is one of the first approaches that exploited correspondences between images to project pixels in real images into a virtual image plane. This approach is similar to view morphing [136] and linearly interpolates the correspondences, or flow vectors, to predict intermediate viewpoints, as Figure 6.3 indicates.

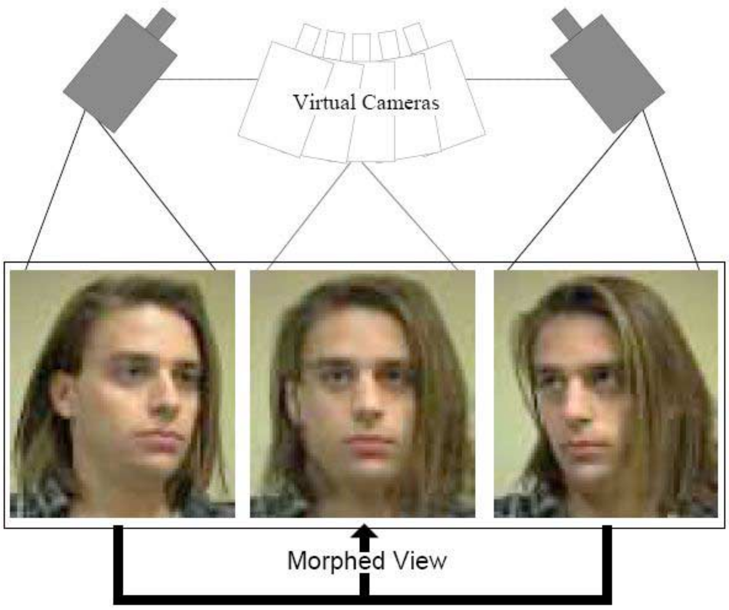

Figure 6.3: View morphing: interpolating behavior in different views.

6.2.4.2 Illumination Compensation

As we know, camera distance and positioning will affect illumination, in the sense that the same object surface can appear to be different from different angles. This is because in general illumination sources are not situated infinitely far away from the scene, e.g., in the case of indoor scenes. In multi-view video environments, the impact of illumination mismatches can be very significant. To alleviate this problem, illumination compensation techniques have been used in the context of motion compensated video coding. Although a few compensation techniques [134][135] are developed and robust to illumination mismatches between views, there is much room for further exploiting correlation across views. Moreover, it would be more interesting to considering VLSI implementation when MVC is standardized in the near future.