Video enhancement : content classification and model selection

Please check the document version of this publication:

Video Enhancement

Content Classification and Model Selection

Hao Hu

Advanced School for Computing and Imaging

Video Enhancement: Content Classification and Model Selection

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus, prof.dr.ir. C. J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op dinsdag 9 februari 2010 om 16.00 uur

door

Hao Hu

geboren te Qichun, China

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. G. de Haan
en
prof.dr.ir. R.H.J.M. Otten

Acknowledgments

It is a pleasure to thank many people who made this thesis possible.

First of all I would like to express my most sincere gratitude to my supervisor Prof. Gerard de Haan, who offered me this Phd position. During these years, he always provided me with helpful suggestions, inspirational advice and constant encouragement and supported me to try out my own ideas. I deeply appreciate his constructive criticism and comments from the initial conception to the end of this work and always feel a great privilege to work with him. I also wish to thank Prof. Ralph Otten for his kind advice and support to my Phd study. Besides my promoters, I would like to thank the rest of my thesis committee: Prof. Jan Biemond, Prof. Hartmut Schröder, Prof. Peter de With, and Prof. Ingrid Heynderickx, for their insightful suggestions and comments.

I would like to give the acknowledgements to the colleagues in the ES group of Technische Universiteit Eindhoven. Many thanks to Marja de Mol-Regels and Rian van Gaalen for their help and support since I started my Phd study. I am very grateful to Meng Zhao for not only sharing his valuable Phd experience in the research work as a senior colleague but also offering a lot of help in everyday life as a sincere friend. Thanks to Sander Stuijk for helping me solve computer problem. I was lucky to have Amir Ghamarian and Chris Bartels as my officemates. I would like to thank them for their help and all the interesting office talks in the last few years. Also I would like to thank my students Aron Beurskens, Yifan He and Yuanjia Du for their contributions to this work.

The research work presented in this thesis has been carried out as the cooperation with Philips Research Laboratory Eindhoven. I would like to express my thanks to the colleagues for their support at Philips Research. I would like to thank the group management Geert Depovere, Hans Huiberts and Ine van den Broek for their support to my research in the group. I would like to thank Paul Hofman for guiding me into the image processing field when I started my internship at the group. I would like to thank Ihor Kirenko, Ling Shao, Jelte Vink, Arnold van Keersop, Justin Laird and Fabian Ernst, for their help and support to my work. Many thanks to group members for spending their times for participating subjective assessment experiments and reviewing my papers.

iii

Summary

The purpose of video enhancement is to improve the subjective picture quality. The field of video enhancement includes a broad category of research topics, such as removing noise in the video, highlighting some specified features and improving the appearance or visibility of the video content. The common difficulty in this field is how to make images or videos more beautiful, or subjectively better. Traditional approaches involve lots of iterations between subjective assessment experiments and redesigns of algorithm improvements, which are very time consuming. Researchers have attempted to design a video quality metric to replace the subjective assessment, but so far it is not successful.

As a way to avoid heuristics in the enhancement algorithm design, least mean square methods have received considerable attention. They can optimize filter coefficients automatically by minimizing the difference between processed videos and desired versions through a training. However, these methods are only optimal on average but not locally. To solve the problem, one can apply the least mean square optimization for individual categories that are classified by local image content. The most interesting example is Kondo's concept of local content adaptivity for image interpolation, which we found could be generalized into an ideal framework for content adaptive video processing. We identify two parts in the concept, content classification and adaptive processing. By exploring new classifiers for the content classification and new models for the adaptive processing, we have generalized a framework for more enhancement applications.

For the part of content classification, new classifiers have been proposed to classify different image degradations such as coding artifacts and focal blur. For the coding artifact, a novel classifier has been proposed based on the combination of local structure and contrast, which does not require coding block grid detection. For the focal blur, we have proposed a novel local blur estimation method based on edges, which does not require edge orientation detection and shows more robust blur estimation. With these classifiers, the proposed framework has been extended to coding artifact robust enhancement and blur dependant enhancement. With the content adaptivity to more image features, the number of content classes can increase significantly. We show that it is possible to reduce the number of classes

without sacrificing much performance.

For the part of model selection, we have introduced several nonlinear filters to the proposed framework. We have also proposed a new type of nonlinear filter, trained bilateral filter, which combines both advantages of the original bilateral filter and the least mean square optimization. With these nonlinear filters, the proposed framework show better performance than with linear filters. Furthermore, we have shown a proof-of-concept for a trained approach to obtain contrast enhancement by a supervised learning. The transfer curves are optimized based on the classification of global or local image content. It showed that it is possible to obtain the desired effect by learning from other computationally expensive enhancement algorithms or expert-tuned examples through the trained approach.

Looking back, the thesis reveals a single versatile framework for video enhancement applications. It widens the application scope by including new content classifiers and new processing models and offers scalabilities with solutions to reduce the number of classes, which can greatly accelerate the algorithm design.

Contents

Chapter 1

Introduction

Video is one of the great inventions of the 20th century. With its rapid growth, it has changed people's life in many ways and after decades of development still keeps bringing new visual experiences. Analog video was first developed for cathode ray tube television systems [1], which has been used for half a century. The evolution to digital video brings rapid advances in video technology, along which several new technologies for video display devices such as liquid crystal display (LCD) [10] and plasma display panel (PDP) [3] have been developed. Standards for television sets and computer monitors have tended to evolve independently, but advances in digital television broadcasting and recording have produced some convergence [7]. Powered by the increased processor speed, storage capacity, and broadband Internet access, computers can show television programs, video clips and streaming media.

In the past televisions used to be the main video platform. The relentless progression of Moore's Law [23] coupled with the establishment of international standards [62] for digital multimedia has created more diverse platforms. General-purpose computing hardware can now be used to capture, store, edit, and transmit television and movie content, as opposed to older dedicated analog technologies. Portable digital camcorders and camera-equipped mobile phones allow easy capturing, storing, and sharing of valuable memories through digital video. Set-top boxes are used to stream and record live digital television signals over broadband cable networks. Smart camera systems provide a peaceful security through intelligent surveillance. The ubiquitous dissemination of digital information in our everyday lives provides new platforms for digital video and generates new challenges for video processing research.

Traditional video enhancement techniques focused on topics such as noise reduction, sharpness and contrast enhancement in processing the analog video signal. Since the advent of digital video and the emergence of more diverse platforms, the traditional techniques have exposed their limitations. The rapid devel-

opment of video technology poses new challenges and asks for new solutions for the increasing video enhancement applications.

In the following, we shall first briefly introduce recent developments in video technology and review some trends in developments of video enhancement techniques. Then we will discuss our research objective and opportunities.

1.1   Developments in video technology

The development of video processing techniques is closely coupled to the video technology. With the advent of digital technology, the video signal can be digitalized into pixels and stored in a memory, which allows easy and flexible fetch and operation on the pixels to achieve more advanced video processing. The digital video signal contains more dimensions of data than other types of signal such as audio. To enable real-time processing, it requires much more processing power to cope with the ever increasing demand for better picture quality, such as higher resolution and frame rate. Therefore, the evolution of video processing system has been dependant on the progress of semiconductor technology and supporting techniques such as displays.

1.1.1   Transition from analog to digital

Until recent decades, video has been acquired, transmitted, and stored in analog form. The analog video signal is a one-dimensional electrical signal of time. It is obtained by a scanning process which includes sampling the video intensity pattern in the vertical and temporal coordinates [6]. Digital video is obtained by sampling and quantizing the continuous analog video signal into a discrete signal. For the past two decades, the world has been experiencing a digital revolution. Most industries have witnessed a change from analog to digital technology, and video was no exception.

Compared to analog video, digital video has many advantages. The digital video signal is more robust to noise and is easier to use for encryption, editing and conversion [6]. The digital video frames are stored in a memory, which provides access to neighboring pixels or frames. For video system design, it also allows first time right design of complex processing. The video processing algorithms can be mapped to a programmable platform and the design time is greatly reduced. These advantages allow a number of new services and applications to be introduced. For example, the TV broadcasting industry has introduced new services like interactivity, search and retrieval, video-on-demand, and high definition television (HDTV) [7]. The telecommunication industry has provided video conferencing and videophones over a wide range of wired and wireless networks [8].

The consumer electronics industry has seen great convenience of easy capturing and sharing of high quality digital video through the fast development of portable digital cameras and camcorders [9].

Although digital video has many advantages, it also shows some problems. Since digital video requires large amounts of bandwidth and storage space, high compression is essential in order to store and transmit it. However, high compression will cause annoying coding artifacts, which brings new challenges to design good coding artifact reduction algorithms.

1.1.2 Developments in display technology

The cathode ray tube (CRT) has been widely used in televisions for a half century since the invention of television [2]. As a mature technology, the CRT has many advantages, like wide viewing angle, fast response, good color saturation, long lifetime and good image quality [1]. However, a major disadvantage is its bulky volume.

(A) TV in 1940   (B) TV in 2009
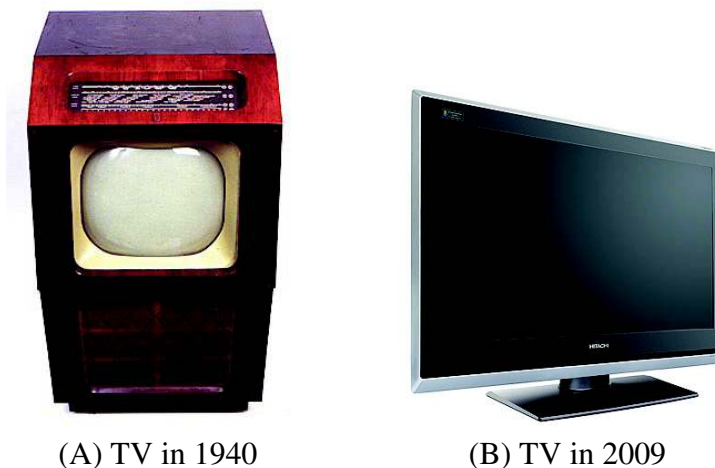
Figure 1.1: The cathode ray tube television in 1940 (A) and the flat panel display television in 2009 (B).

Flat panel displays with a slim profile like liquid crystal display (LCD) and plasma display panel (PDP) are developed to solve the problem [14][4]. Besides the slim profile, the flat panel display has many other advantages over the CRT, such as higher resolution and no geometrical distortion. The rapid development of flat panel displays [15] has made larger panels with a more affordable price. Nowadays these display technologies have already replaced the CRT in the television market. Nevertheless, these flat panel display technologies are not perfect.

For example, PDP tends to have false contours [17] and the sample-and-hold effect of LCD causes motion blur [18]. The imperfections of these display technologies have led to the development of flat panel display signal processing [19]. Next generations of flat panel display technologies like Organic light-emitting diode (OLED) [20] and not-yet released technologies like Surface-conduction Electron-emitter Display (SED) or Field Emission Display (FED) [21] are predicted to replace the first generation of flat display technologies.

Compared to conventional ways of receiving information, such as books and newspapers, electronic displays such as televisions and monitors have a constraint that they typically have to be fabricated on glass substrates. Flexible flat panel displays [22] that can be rolled as papers as shown in Fig. 1.2 are emerging. Flexible displays are thin, robust and lightweight and indicate the future direction of the display technology.

Figure 1.2: Philips flexible display.

As these flat panel displays show a nearly perfect and sharp picture, any imperfections in the video, such as coding artifacts, may become more visible. This urges the needs for developing high quality video enhancement algorithms.

1.1.3   Developments in processing platforms

Moore's law [23] predicts that the number of transistors that can be placed inexpensively on an integrated circuit will increase exponentially, doubling approximately every two years as shown in Fig. 1.3. The vigorous development in video

Figure 1.3: Transistor count in processors 1997-2008 and Moore's law. Source [24].

technology was enabled by the rapid technological progress reflected by Moore's law. With the restless pursuit of faster processing speed, higher resolution and frame rate, and higher memory capacity, the demand for processing power is increasing exponentially every year. The advances in semiconductor technology predicted by Moore's law has successfully met the increasing demand for computing power.

Application-specific integrated circuit (ASICs) is the first hardware platform for video processing. It is designed for a specific purpose and the design can more be easily optimized so that it usually provides better total functionality and performance. As the applications become more complex, the ASIC design or change takes longer time and the percentage of first time right design decreases. This leads to a higher implementation cost. Therefore, a programmable hardware

platform which allows late software modification starts to be used for video processing.

One of the earliest programmable hardware platform is the computer system used by NASA to process the video taken in the space [115]. Since then, different programmable processing hardware platforms have been developed. Due to the inherent parallelism in the pixel operation for common video processing applications, architecture concepts, such as single instruction multiple data (SIMD) and very long instruction word (VLIW) [25], were built to be massively parallel in order to cope with the vast amounts of data in the general purpose processors (GPPs).

Although GPPs have massive general-purpose processing power, they are extremely high power-consuming large devices requiring about one hundred watts. The need for application-specific hardware with a smaller size has led to the development of digital signal processor (DSP) and field programmable gate array (FPGA) in the 1990s [26]. In recent decades, further development has led to the video processing architecture of application-specific instruction-set processors (ASIPs) which combines the advantages of ASICs and GPPs, and eventually ASIPs have brought all the necessary computing power and flexibility for real-time image/video processing onto a single chip.

The ASIP approach has found the right balance between efficiency and flexibility and is promising for the next generation of video processing hardware architecture. For video enhancement algorithms, it is also desirable to have a single software architecture, which not only offers high performance as dedicated solutions but also is applicable for a wide range of applications.

1.1.4   Developments in application domain

Alongside the developments in hardware architectures for image/video processing, there have also been many notable developments in the application of video processing. Recently the development of smart camera systems [33][34] have become a hot topic of research worldwide. Relevant technologies used in the consumer equipments include automatic adjust of focus [35][78] and white balance [36]. In digital video surveillance system, there has been an increasing number of more advanced technologies, such as robust face detection and recognition [40][41][42], gesture recognition[37], human behavior analysis [39], and distributed multiple camera network [38]. In the endless pursuit for a perfect picture, research in developing high quality algorithms for processing videos obtained by consumer digital cameras, such as super resolution [28][29], high dynamic range imaging [30], and texture synthesis techniques [31][32]. Such techniques have received considerable attention and are expected to progress in the future.

Recent years also see the convergence of multiple applications towards a single device. In the past, consumers had many individual portable electronic devices to meet their needs for entertainment, information, and communication: a mobile phone for communication, a digital camera for pictures, an MP3 player for listening to music, a portable game console for playing games, and a notebook computer for email and Internet surfing. However, with the introduction of the multi-function portable electronics such as iPhone as shown in Fig. 1.4, consumers now have the option of combining these technologies into a single device.

For the emerging applications in different video platforms and the convergence of these applications, a scalable approach with a single architecture for video enhancement algorithms is preferred.
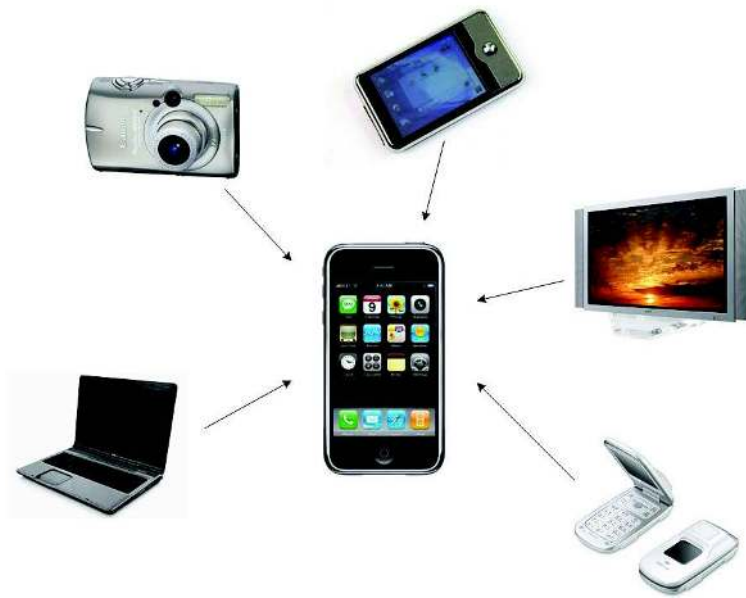
Figure 1.4: The convergence of multiple applications towards a single device: iPhone example.

1.2 Content-adaptivity in video enhancement

Since the invention of video, video enhancement has been a very important part of video technology. Video enhancement consists of a broad category of techniques to increase the video quality, such as removing noise in the video, highlighting some specified features and improve the appearance or visibility of the video content. Looking at the developments of these video enhancement techniques, we see that there are trends towards more and more detailed content adaptivity, from

non-adaptivity to adaptivity, from global to more local image properties. In this section, we will introduce such trends in some common video enhancement applications, including noise reduction, image interpolation and contrast enhancement.

1.2.1   Content-adaptivity in noise reduction

First we see the content adaptivity trend in noise reduction, which is one of the most common video enhancement techniques. Early methods to remove noise generally filter the video with a low pass filter which is a smoothing operation. Usually the smoothing operation is done by setting the output pixel to the average value, or a weighted average of the neighboring pixels [115]. The strength of the smoothing can be adjusted according to the average noise level in the image. Since the smoothing operation is uniformly applied on the entire image, they have good performance at eliminating noise at flat areas. However, they also blur the signal edge. In order to solve the problem, algorithms such as coring [45] that can adapt themselves to the signal amplitude have been introduced. They have a stronger smoothing effect at flat areas and a less smoothing effect at detailed areas to preserve signal edges. Further progress in noise reduction algorithms has brought more adaptivity to local content such as image edge and structures [127]. Adaptive filters based on local edge or structure information can have a better performance at reconstructing image details from noisy input. Fig. 1.5 shows example results of different noise reduction techniques, from non-adaptivity to adaptivity and from coarse adaptivity to detailed adaptivity. Clearly more detailed adaptivity brings more performance improvement.

1.2.2   Content-adaptivity in image interpolation

Similar trends towards content adaptivity can be also found in the development of image interpolation techniques. Image interpolation is concerned about displaying an image with a higher resolution, while achieving the maximum image quality. This has been traditionally approached by linear methods, which use the weighted sum of neighboring pixels to estimate the interpolated pixel. Because the linear methods use a uniform filter for the entire image without any discrimination, they tend to produce some undesired blurring effects in the interpolated images [95].

Some content-adaptive methods have been introduced to solve the problem [97][103][102]. One category of these content-adaptive methods can be labeled as edge directed methods. Unlike the linear methods which use a uniform weight setting, they are designed to detect the edge direction and apply more optimal weighting to pixel positions along the edge direction as shown in Fig. 1.6. Therefore, better interpolation performance is achieved at the edges. Besides edge-directed
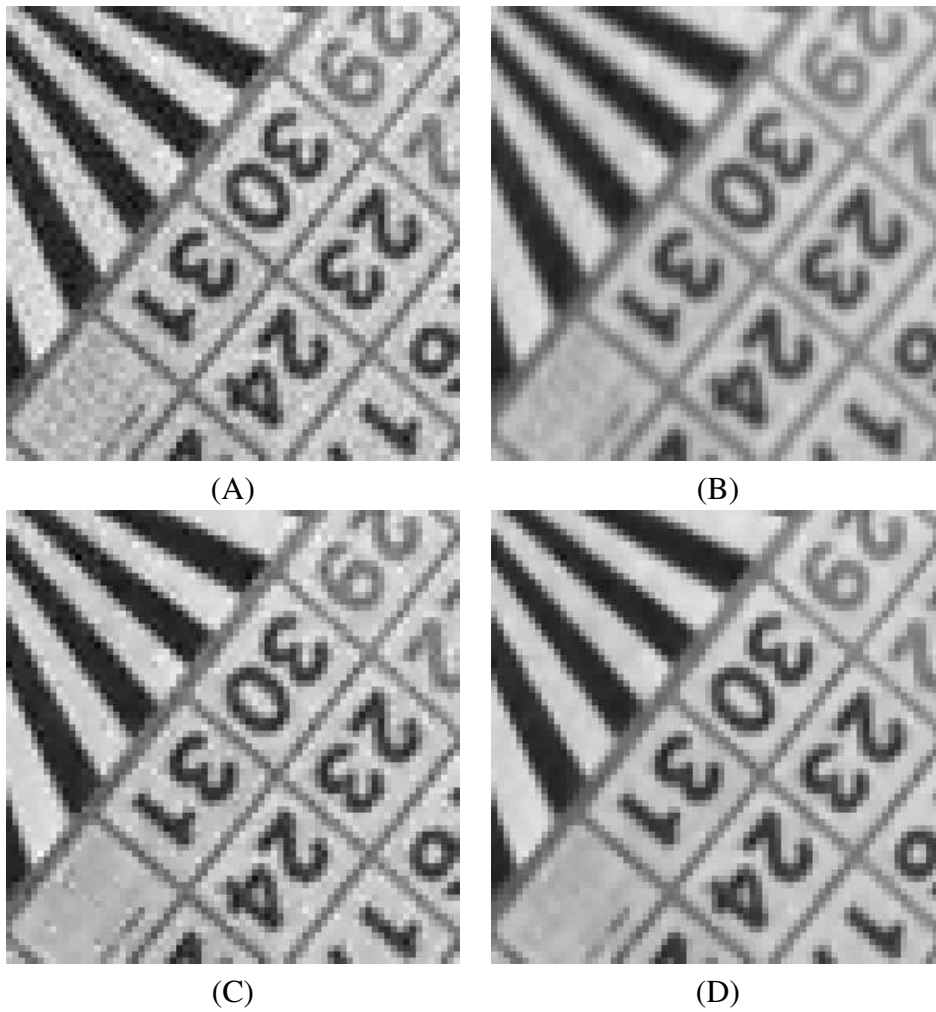
Figure 1.5: Content adaptivity in noise reduction: (A) noisy input, (B) filtered by a single filter, (C) filtered by a coring algorithm, (D) filtered by structure adaptive filters.

methods, some classification-based methods, which depend on more general image structures than edges, have been proposed by Kondo [97] and Atkins [105]. The classification-based methods use a pre-processing step to classify the image block into a number of classes as shown in Fig. 1.7. Then the image block can be interpolated using a linear filter that is optimized for the class. These content-adaptive methods prove to have better performance on specific image structures such as edges, than standard linear methods, such as bi-linear and bi-cubic interpolation.

weight

1/4   1/4   1/4   1/4

(A) Linear interpolation

1/2   0   0   1/2

0   1/2   1/2   0

(B) Edge directed interpolation
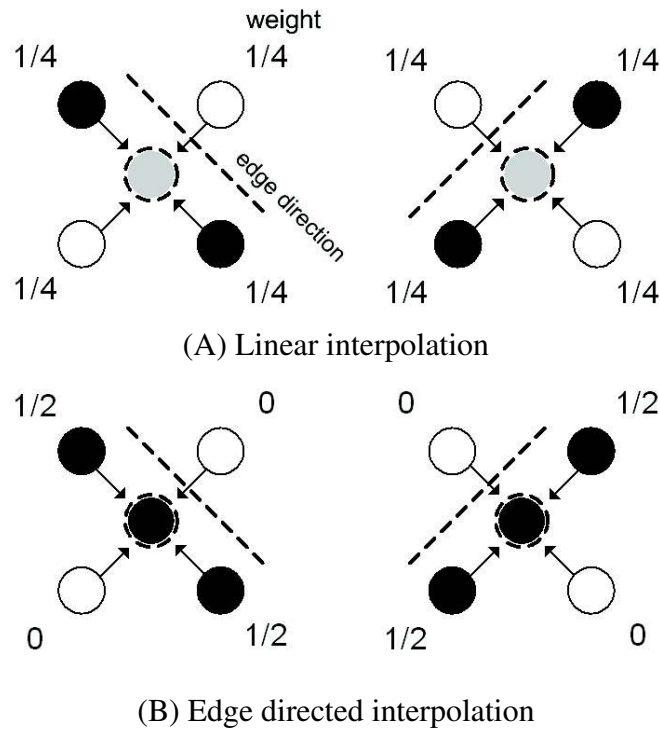
Figure 1.6: Image interpolation: the central pixel value to be interpolated is determined by the weighted sum of the neighboring pixel values. (A) linear interpolation: uniform weight setting regardless of image content, (B) edge directed interpolation: assigning more weight to the pixels along the edge direction.

1.2.3 Content-adaptivity in contrast enhancement

Without exceptions, there are also trends to content-adaptivity in the development of contrast enhancement. Contrast enhancement is usually done with a grey-level transfer curve. The transfer curve maps a pixel value in an input image to a pixel value in the processed image. Typically the values of the transfer curve are stored
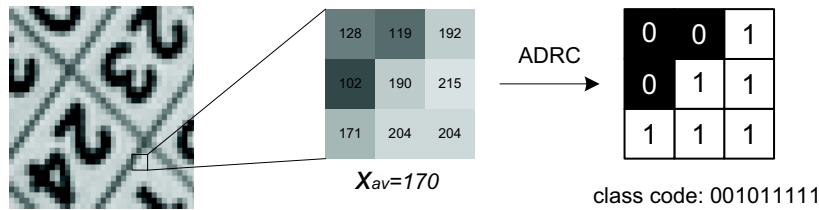
Figure 1.7: Image structure classification proposed by Kondo: the pixel values in a local aperture are compared with the average pixel value within the aperture. The result is a binary code which represents the structure pattern.

in one-dimensional array and the mappings are implemented by look-up-tables [113]. Early grey-level transformations use some basic type of pre-defined functions for image enhancement, such as linear and logarithmic functions [114]. Fig. 1.8 shows an example of contrast stretch with a piece-wise linear transfer curve. Another example of gamma correction is shown in Fig. 1.9. These transfer curves are fixed for the entire image regardless of the change of image content.

(A)   (B)   (C)
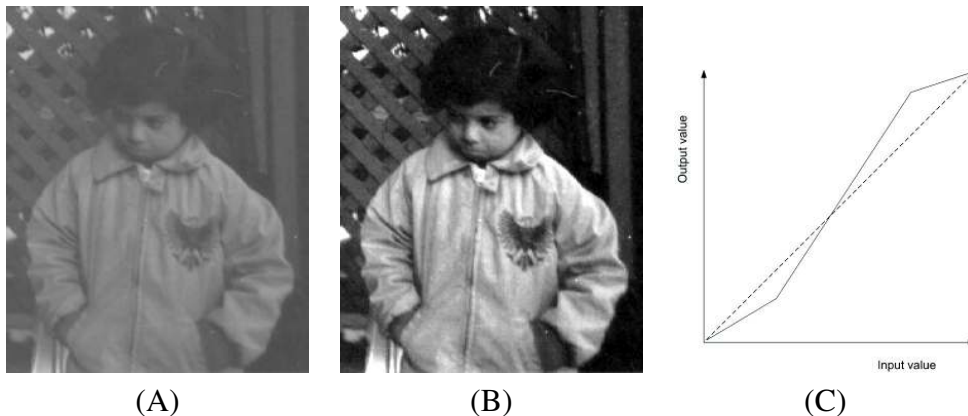
Figure 1.8: Contrast stretch by a piece-wise linear transfer curve: (A) input image, (B) contrast stretched version, (C) the piecewise linear transfer curve.

Further development of contrast enhancement algorithms has proposed to calculate a transfer curve depending on a histogram of the image content. In these approaches, the transfer curve depends on the histogram of the entire image. One typical example is histogram equalization, which re-maps grey scales of the image such that the resultant histogram approximates that of the uniform distribution [117]. Content adaptivity to the entire image may not be optimal since the local image content can change from one region to another in an image. Therefore,
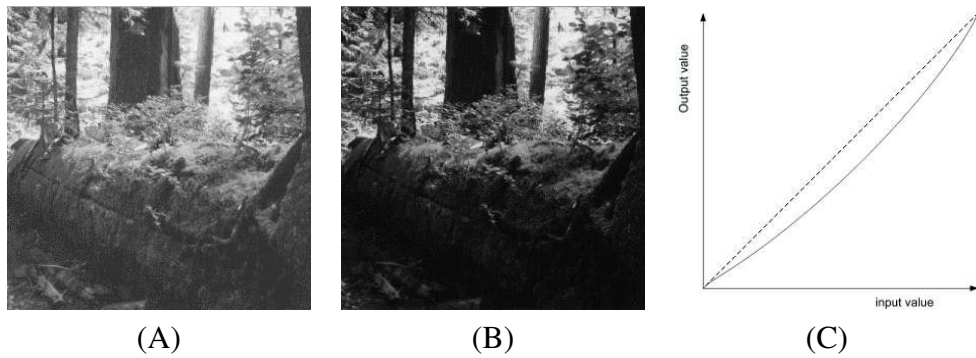
Figure 1.9: Gamma correction: (A) input image, (B) after gamma correction, (C) the transfer curve for gamma correction.

local content adaptive contrast enhancement algorithms [120] [118] have been proposed to improve the local enhancement performance. These algorithms find transfer curves for different regions based on its neighborhood content. Fig. 1.10 (B) shows the result of histogram equalization which adapts a transfer curve to its global content. And Fig. 1.10 (C) shows an example of local contrast enhancement where the local contrast has been enhanced based on the local content.
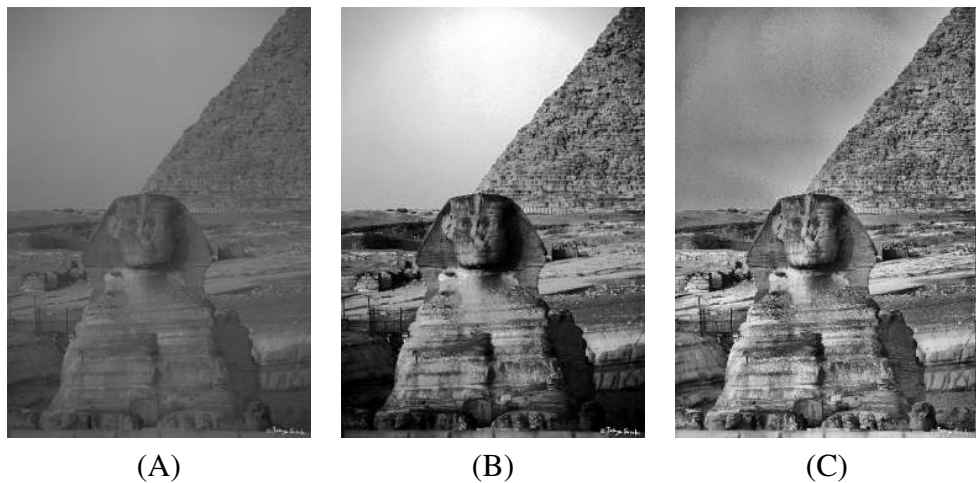
Figure 1.10: Adaptive contrast enhancement (A) input image, (B) result from the global adaptive contrast enhancement, (C) result from the local adaptive enhancement.

1.3 Research objective and opportunities

1.3.1 Research goal

Although video enhancement usually consists of quite diverse topics, such as sharpness, contrast, color and resolution improvement, and noise reduction, the common ultimate goal of video enhancement is to improve the subjective picture quality [44]. How to achieve the goal is not always trivial. Traditional approaches involve lots of iterations between subjective assessment experiments and redesigns of algorithm improvements as shown in Fig. 1.11, which are very time consuming. For decades researchers have been trying to design a video quality metric to replace the subjective assessment, but so far these attempts have not been successful. The mean square error (MSE) is often used as a metric to measure the difference between image outputs and ideal versions. However, the MSE metric only reflects the image quality *on average* not locally. The optimal filter for an edge, for instance, differs from the optimal filter for a flat area as suggested in Fig. 1.6. Therefore, processing which is optimal on average is likely to be sub-optimal locally. To achieve locally optimal processing, it is important to include local content-adaptivity into the least mean square optimization. The most interesting example is Kondo's concept of local content adaptivity [97] as it offers a nice, generally applicable framework. Kondo's method classifies local image content into a number of classes and in every class a dedicated LMS optimal filter is used for adaptive filtering.
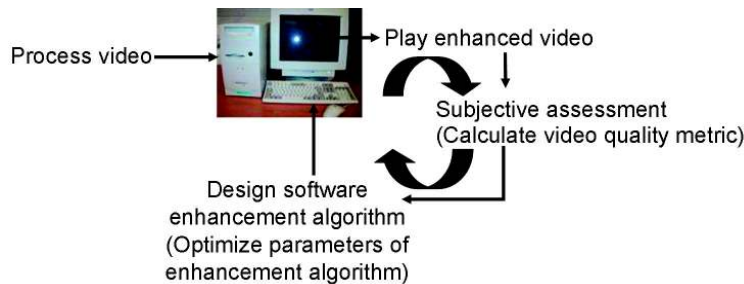
Figure 1.11: The traditional approach to design enhancement algorithms: it has to iterate between subject assessment experiments and algorithm redesigns. Attempts to design a video quality metric to replace the time consuming subject assessment are not successful so far.

We identify two parts in this concept, content classification and processing model selection, which could be further generalized for a broader range of video enhancement application.

In the content classification part, previous work has been only focused on local structure classification. Exploration of other classifiers could be beneficial

for many applications. As the number of classes will increase exponentially with more included classifier, it would be desirable to achieve simplication by reducing the number of filters without serious performance loss.

In the processing model selection part, although the linear LMS filter is always used as the processing model and usually has a satisfactory result even though it is not designed for different types of processing, a dedicated design is expected to yield more effect. For contrast enhancement, it is also not clear how to apply this training approach, but would provide an interesting application.

To generalize this concept by incorporating more new classifiers and new types of processing models is considered to be of great importance and since it is expected to lead to synthesis of designs with an improved cost-performance and reduced design time.

In conclusion, we aim this PhD-study at proposing (synthesizing) new classifiers and new models towards a generalized content adaptive processing framework for digital video enhancement, while keeping complexity at a reasonable level.

1.3.2 Opportunities

Our research work starts with Kondo's method [97] for image interpolation, which has been extended later as the structure-controlled LMS filter for other resolution enhancement application such as de-interlacing [47][48]. The steps in Kondo's method are described as follows. First the local content of the input video is classified by local structure such as using adaptive dynamic range coding as shown in Fig. 1.7, into a number of different content classes. Then in each class, a trained linear filter is used as shown in Fig. 1.12. The output pixel y_c is calculated as:

$$y = W_c^T X \tag{1.1}$$

where W_c is the coefficient vector for class c and X is the input pixel vector which belongs to class c.

The look-up-table is obtained through an off-line training for individual classes. In a training procedure as shown in Fig. 1.13, original high resolution images are used as the desired reference and then are down-scaled as the simulated input. Before training, the input and reference image data are classified using ADRC on the input vector. The pairs that belong to one specific class are used for training, resulting in optimal coefficients for that class. The coefficient vector W_c is optimized by minimizing the mean square error between the output y_c and the desired version d_c. The mean square error MSE is:

$$MSE = E[(y_c(t) - d_c(t))^2] = E[(W_c^T X_c(t) - d_c(t))^2]. \tag{1.2}$$
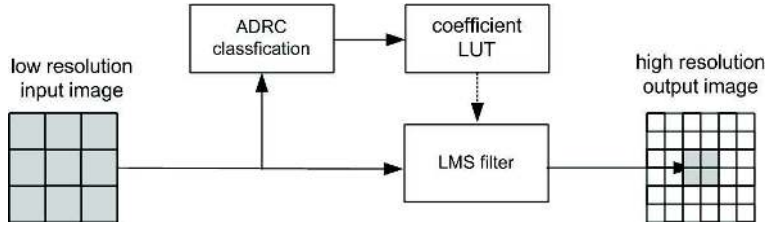
Figure 1.12: Kondo's method for image interpolation. The input pixel vector from a local window first is classified by the adaptive dynamic range coding. Then the LMS filter coefficients are fetched from a look-up-table. The high resolution pixel is the output of the LMS filtering.

Taking the first derivative with respect to the weights and setting it to zero, the coefficients W are obtained:

$$W_c^T = E[X_c X_c^T]^{-1} E[X_c d_c].$$ (1.3)
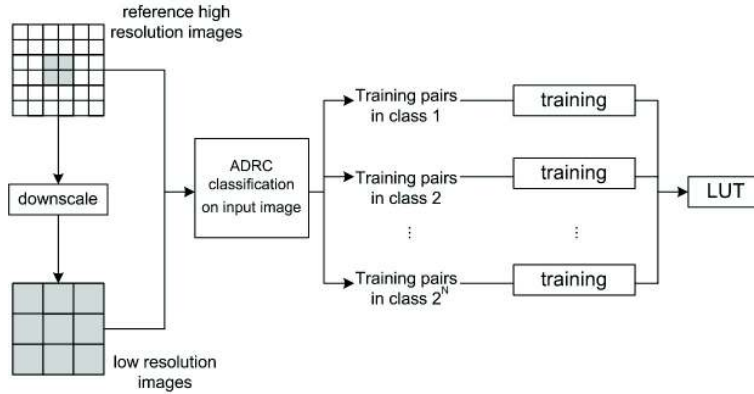
Figure 1.13: Kondo's method to obtain optimal filter coefficients: Original high resolution images are down-scaled to generate the simulate input and reference images for the training. The filter coefficients are optimized for individual classes and stored in the look-up-table.

If we look at the concept of Kondo, it consists of two parts, structure classification and LMS filtering, as shown in Fig. 1.14. We can further extend the structure adaptive LMS filtering to a more general framework of content adaptive video processing as shown in Fig. 1.15. Then the corresponding two parts become content classification and adaptive processing. We expect plenty of opportunities to include more ingredients into these two parts to increase the performance and thus widen the application scope of the framework.
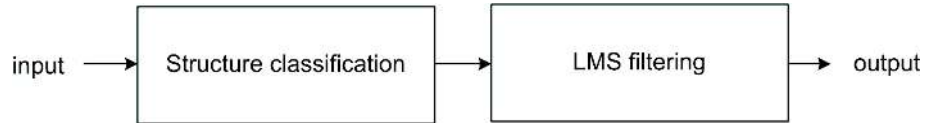
Figure 1.14: Structure controlled LMS filter.
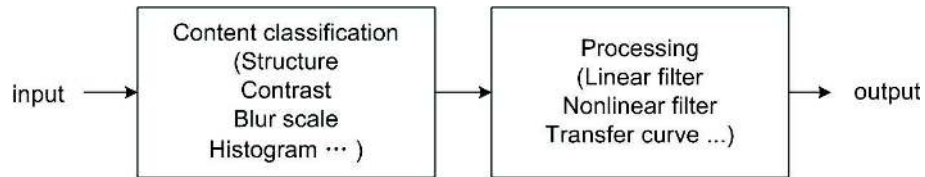
Figure 1.15: Generalized content adaptive processing.

A first opportunity is seen to include a coding artifact classifier into the content classification part. This could be designing a classifier to distinguish coding artifacts from real image structure regardless of the compress codec. Previous approaches tried to use local structure and block grid position information. However reliable detection may be difficult for signals compressed by methods with a variable transform block size such as AVC/H.264 [62].

We see another opportunity to include a focal blur estimator into the classification part. Focal blur is another type of image degradation which often occurs in the videos. How to estimate local blur is a challenge. With accurate local blur estimation, one can remove the blur and restore the resolution. Blur dependant video enhancement can be also interesting.

For the adaptive processing part, the filter so far has always been linear. From the literature it is known that nonlinear filters such as rank order filters and bilateral filters may perform better in smoothing tasks where edge preservation is important [90][57]. Also bilateral filters have the ability to locally adapt the filtering to the image content [57]. It is interesting to explore if and how these nonlinear processing modules could be used in our proposed framework, and see how they could improve the enhancement results.

Finally, the content adaptive processing framework always applies filtering. In applications such as contrast enhancement, transfer curves are often used instead of filtering. We will also explore the opportunity to apply the framework to contrast enhancement by using a content adaptive transfer curve in the adaptive processing part.

1.4 Contributions

Based on the research objective, our research has generated the following contributions in the two parts of the content adaptive video processing framework.

1.4.1 Contributions to content classification

Our first contribution in this part is the introduction of a new simple and efficient coding artifact classifier. The two orthogonal image properties, local structure and contrast, are proposed to distinguish real image structure and coding artifacts. Furthermore the distribution of the occurrence of classes can be used for the region quality indication. Based on the classifier, we propose video enhancement algorithms which integrate sharpness and resolution enhancement. This contribution has resulted in a patent application [134] and publications in the Proceedings of IEEE International Conference on Consumer Electronics [129] and International Conference on Image Processing in 2007 [131].

Our second contribution in this part is that we propose a new local blur estimation method that generates consistent blur estimates for objects in an image. First, a novel local blur estimator based on edges is introduced. It uses a Gaussian isotropic point spread function model and the maximum of difference ratio between the original image and its two digitally re-blurred versions to estimate the local blur radius. The advantage over alternative local blur estimation methods is that it does not require edge detection, has a lower complexity and does not degrade when multiple edges are close. With the blur estimates from the proposed blur estimator and other clues from the image, like color and spatial position, the image is segmented using clustering techniques. Then within every segment, the blur radius of the segment is estimated to generate a blur map that is consistent over objects. The result has led to a patent application [135] and publications in the Proceedings of IEEE International Conference on Image Processing in 2006 [128] and International Conference on Advanced Concepts for Intelligent Vision Systems in 2007 [132].

Our third contribution - A major problem of content adaptive filtering is that, with the increasing number of features, it can have an impractically large number of classes, many of which may be redundant. For hardware implementation, a class-count reduction technique that allows a graceful degradation of the performance would be desirable. We propose three options, which use class-occurrence frequency, coefficient similarity and error advantage, to reduce the number of classes. The results show that with the proposals the number of classes can be greatly reduced without serious performance loss. This contribution has been published at the Proceedings of IEEE International Symposium on Consumer Electronics in 2009 [133].

1.4.2   Contributions to model selection

In the content adaptive processing applications using the proposed framework in the previous research, the processing model part has always been a linear filter. To further improve the performance, we extend the model to include nonlinear filters, such as the rank order filter, the hybrid filter and the neural network. Additionally we propose a new type of nonlinear filter, the trained bilateral filter. The trained bilateral filter adopts a linear combination of spatially ordered and rank ordered pixel samples. It possesses the essential characteristics of the original bilateral filter and the ability to optimize the filter coefficients to achieve desired effects. This contribution has resulted in a patent application [136] and publications in the Proceedings of SPIE Applications of Neural Networks and Machine Learning in Image Processing conference in 2005 [126], SPIE Visual Communications and Image Processing in 2006 [127] and IEEE Conference on Image Processing in 2007 [130].

Furthermore we introduce the proposed content adaptive processing framework to contrast enhancement. We propose a trained approach to obtain the optimal transfer curve for contrast enhancement, which is based on a histogram classification. A training is applied to optimize the transfer curve from a version enhanced by computationally intensive algorithms. Furthermore, we propose a combined global and local contrast enhancement approach using separately trained transfer curves. A global transfer curve and a local one are used to transform the local mean and the difference between the local mean and the processed pixel, respectively. The advantage is that it can adapt to both global and local content and offer optimized enhancement.

1.5   Thesis outline

Besides the introduction and conclusion chapters, this thesis consists of two parts based on the content adaptive video enhancement framework, which can be identified as: video content classification and filter model selection. Fig. 1.16 illustrates the structure of chapters in this thesis. Chapter 2, 3 and 4 show our contribution to the part of content classification. Chapter 5 and 6 show our contribution to the part of model selection. The content of each chapter is summarized as follows.

Chapter 2 presents a novel classifier for coding artifacts, which is based on the combination of local structure and contrast, which does not require coding block grid detection. The good performance of the enhancement algorithm based on the classifier shows the effectiveness of the classifier at distinguishing the coding artifacts. With the help of the coding artifact classification, we are able to build up coding artifacts reduction algorithms combined with resolution up-conversion
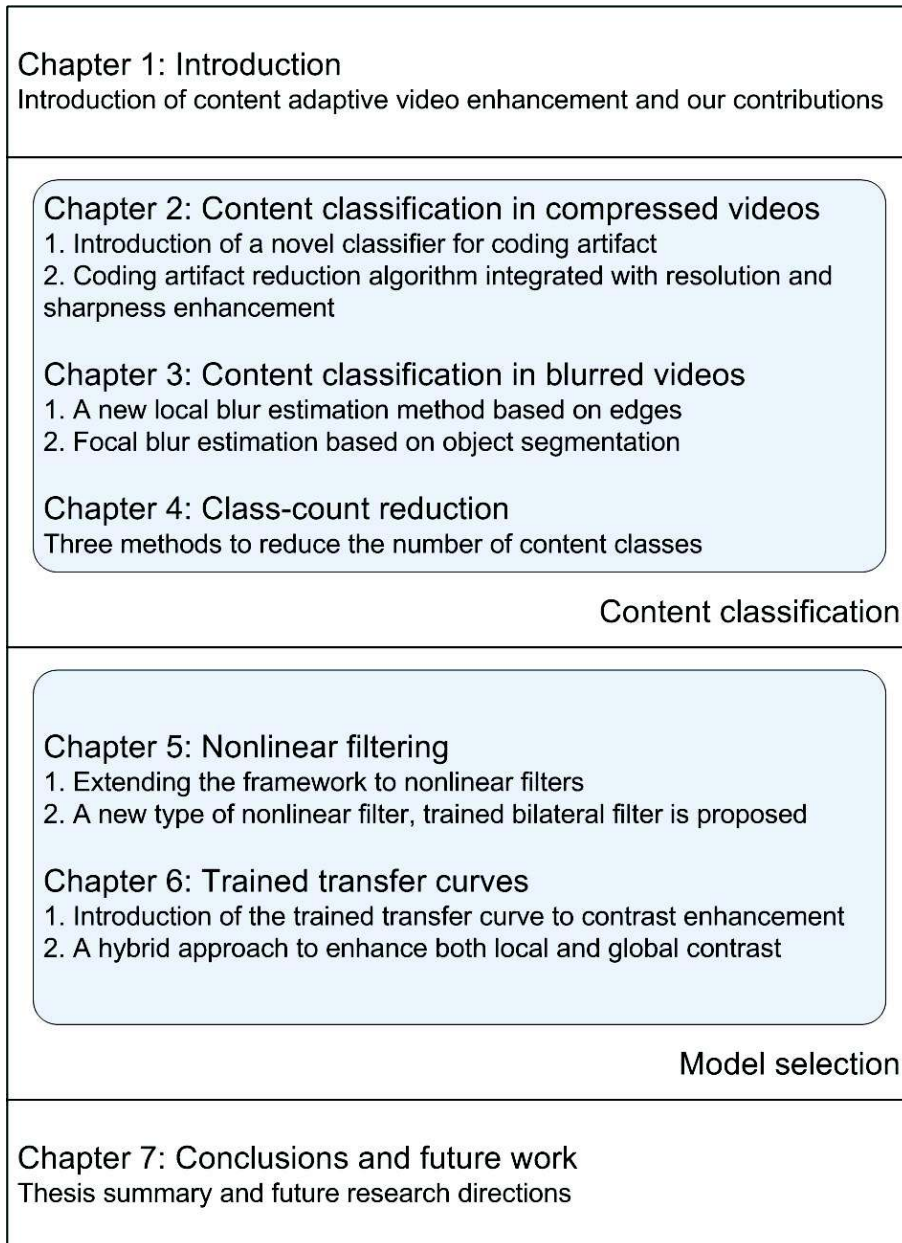
Figure 1.16: Thesis outline.

and sharpness enhancement.

In Chapter 3, we first propose a novel local blur estimation method based on edges, which does not require edge orientation detection and shows more robust estimation than the state-of-the-art method. Then a novel object-based blur estimation approach is proposed to generate a more consistent blur map, which is used to improve the performance of content adaptive enhancement applications such as focus restoration and blur dependent coding artifact reduction.

In Chapter 4, we propose three class-count reduction techniques, class-occurrence frequency, coefficient similarity and error advantage for the content adaptive filtering framework. In the applications of coding artifact reduction and image interpolation, we show that these techniques can greatly reduce the number of content classes without sacrificing much performance.

In Chapter 5, we introduce several types of nonlinear filters for the content adaptive processing framework. Inspired by the bilateral filter and the hybrid filter, we propose a new type of nonlinear filter, the trained bilateral filter. It utilizes pixel similarity and spatial information, as the original bilateral filter, but it can be optimized to acquire desired effects using the least mean square optimization.

Chapter 6 presents a proof-of-concept for the trained approach to obtain contrast enhancement. In this case, a transfer curve depends on the classification of the local and global input image content. Furthermore, a hybrid enhancement method is introduced. The input image is divided into a local mean part and a details part by using the edge-preserving filtering to prevent the halo effect. The local mean part is transformed using a trained global curve based on the histogram classification, and the details part is transformed by a separately trained local curve based on the local contrast classification.

Finally, Chapter 7 summarizes the thesis and points out the future directions.

Chapter 2

Content classification in compressed videos

The goal of the thesis is to generalize the content adaptive filtering framework. In this chapter, we shall focus on the content classification part of the proposed framework to extend the application area to coding artifact reduction.

Coding artifacts often occur in compressed videos when a high compression ratio is used in the compression. They not only degrade the perceptual image quality, but also cause problems to further enhancement in the video processing chain. For example, coding artifacts will become more visible after sharpness enhancement. Therefore, it is essential to detect and reduce coding artifacts before enhancing the compressed video, or ideally to integrate artifact reduction and sharpness enhancement.

Many methods have been proposed to reduce coding artifacts. However, most of them require the compression parameter or the bit stream information to obtain satisfactory results. This is not available in most applications where different standards are used for the compression. For the content adaptive processing framework, designing a coding artifact classifier to the content classification part would lead to solutions for enhancing compressed video. How to design a classifier which can detect coding artifacts for different applications is still a challenge. Furthermore, the enhancement of digital video usually includes sharpness and resolution enhancement. How to combine them as a system solution is also unclear.

To answer these questions, in this chapter, we propose a novel coding artifact detection method, which uses the combination of the local structure and contrast information. Based on the detector, we shall show that coding artifacts in different compression standards can be nicely removed by using the proposed framework. Additionally, we propose a combined approach to integrate sharpness and resolution enhancement. They shall show superior performance in the evaluation part of this chapter.

The rest of the chapter is organized as follows. We start with a brief introduction of different coding artifact reduction techniques in Section 1. Then we propose and analyze the novel coding artifact reduction method in Section 2. In Section 3, we propose a coding artifact reduction method using the proposed coding artifact classification in the framework and compare it with other state-of-the-art methods for different compress standards. Furthermore, the applications to integration of sharpness and resolution enhancement are presented respectively in Section 4 - 5. Finally, we draw our conclusion in Section 6.

2.1   Introduction

With its rapid development, digital video has replaced analog video and has become an essential part of broadcasting, communication and entertainment area in recent years. Consumers are enjoying the convenience and high quality of digital video. On the other hand, digital video also shows some problems. Compared to analog signals, digital signals in general and digital videos in particular require large amounts of bandwidth and storage space. In order to store and transmit it, a high compression is essential. High compression ratios can be achieved by using coarse quantization to less important transform coefficients. However, annoying artifacts may arise as the bit rate decreases. They even become more visible when the digital video is enhanced.

Recently many international coding standards such as MPEG 1/2/4 [62], which all adopt the block-based motion compensated transform, have been successively introduced to compress digital video signal for digital broadcasting, storage and communication. One of the most noticeable artifacts generated by these standards is the blocking artifact at block boundaries. It results from coarse quantization and individual block transformation [54]. On the other hand, due to imperfect motion compensated prediction and copying interpolated pixel from possibly deteriorated reference frames, the blocking artifacts also occur within the block. Additional other artifacts such as ringing and mosquito noise [54] appear inside the coding block as well.

Many methods have been proposed to reduce the blocking artifacts in the literature. According to the domains in which these methods are applied, they could be classified into the following three categories: (1) methods in the spatial domain, (2) methods in the transform domain and (3) iterative regularization between both domains.

The methods in the spatial domain are usually more popular as they do not require DCT coefficients, which are usually not available after decoding. Early approaches such as [51] show that the Gaussian low-pass filter with a high-pass frequency emphasis gives the best performance. Reeves [52] proposed to apply

the Gaussian filter only at the DCT block boundary. Such methods usually examine the discontinuity at the block boundary and then apply low-pass filtering to remove possible artifacts. Block boundary position information is required for these methods and reliable detection may be difficult for videos compressed by compression methods with a variable transform block size such as in H.264 [62] or in case of position dependent scaling. In order to alleviate the blocking artifacts not only at the block boundary but also those inside the block, an in-loop filter which is inside the encoder loop has been adopted in the H.264 standard. The in-loop filtering is applied on every single frame after it gets encoded, but before it gets used as reference for the following frames. This helps avoiding blocking artifacts, especially at low bit rates, but will slow down en/decoding. The sigma filter [86] and the bilateral filter [57] [100] have also been reported to have good results at removing coding artifacts including ringing and mosquito artifacts.

The second category includes approaches that try to solve the problem of artifact reduction in the transform domain. The JPEG standard [53] introduced a method to reduce the block discontinuities in smooth areas of a digital coded image. The DC values from current and neighboring blocks are used to interpolate the first several AC coefficients of the current block. Minami [55] proposed a criterion, the mean squared difference of slope (MSDS), to measure the impact of blocking effects. In his method, the coefficients in the DCT transform domain are filtered to minimize the MSDS. This approach was followed by Lakhani [56] for reducing block artifacts. In their approaches, the MSDS is minimized globally and four lowest DCT coefficients are predicted. The disadvantage of such methods is, they can not reduce the blocking artifacts in the high frequency area. As another approach in the transform domain, Nosratinia [63] proposed a JPEG de-blocking technique by reapplication of JPEG compression. The algorithm uses JPEG to re-compress shifted versions of a compressed image. By averaging the shifted versions and the input image, the resulting artifact reduced image is obtained.

The third category includes methods that are based on the theory of projection on convex sets (POCS). In these POCS-based methods [64] [65] [66], closed convex constraint sets are first defined to represent all knowledge on the original uncompressed image. For instance, one set could represent the quantization range in the DCT transform domain and another set could represent the band-limited version of the input image, which does not contain the high frequency possibly caused by the artifacts. Then, alternating projections onto these convex sets are iteratively computed to recover the original image from the coded image. These POCS-based methods are effective at removing blocking artifacts. However, they are less practical for real time applications, because the iterative procedure increases the computation complexity.

Although there have been a wide range of coding artifact reduction methods available, most of them require the compression parameter or the bit stream in-

formation to obtain good performance. However, this information is usually not available for many applications where the input source can be compressed by different compression standards. Therefore, a post-processing algorithm that can detect coding artifacts on spatial data and further reduce different types of coding artifacts is essentially needed. The proposed content adaptive processing framework could provide such a solution if a new classifier for coding artifact detection can be designed and included in the framework.

2.2 Coding artifact detection

As we can see in the previous section, many coding artifact detection methods rely on the DCT block position information, which means detection of the DCT block position is required. Few methods attempt to detect coding artifacts regardless of the block position. In this section, we will explore how to detect coding artifacts through image content analysis without any compression information.

Due to the block based compression, coding artifacts usually manifest themselves as distinguishable luminance patterns, for instance, blocking artifacts show a pattern of horizontal and vertical edge. For that reason, we continue to use the adaptive dynamic range coding (ADRC) proposed in Kondo's concept [97] to classify the local structure and hope this information can help distinguish coding artifacts. As shown in Fig. 1.7, the 1-bit ADRC code of every pixel is defined by:

$$ADRC(x_i) = \left\{ \begin{array}{ll} 0, & if \ x_i < x_{av} \\ 1, & otherwise \end{array} \right. \qquad (2.1)$$

where x_i is the value of pixels in the filter aperture and x_{av} is the average pixel value in the filter aperture. We use a diamond shape filter aperture suggested in Fig. 2.2 to balance between the performance and the complexity.

As the coding artifacts occur after compression, we measure the changes of the ADRC class occurrence frequency in a set of randomly selected video sequences before and after compression. Here, the occurrence frequency of a class means how many times that class has occurred in the sequences. Fig. 2.1 shows the ADRC patterns of the first ten classes with the largest absolute increase in the occurrence frequency after compression in five randomly selected test sequences. We could find some common ADRC classes shown in Fig. 2.2. It indeed seems likely that pixels belonging to these ADRC classes can be coding artifacts.

In order to evaluate the effectiveness of using the ADRC classification to distinguish coding artifacts, we propose to simply use the common ADRC classes shown in Fig. 2.2 as a "coding artifact detector" and measure the detector's performance using a "ground truth" map which indicates which pixels are coding artifacts. The difference between the compressed image and its uncompressed

	ADRC patterns of the first ten classes with the largest absolute occurrence frequency increase
Sequence A	
Sequence B	
Sequence C	
Sequence D	
Sequence E	

Figure 2.1: The ADRC patterns of the first ten classes with the largest occurrence frequency increase after compression in five randomly selected sequences.
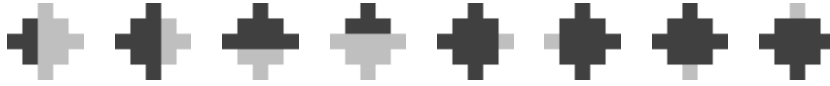
Figure 2.2: Artifact-alike classes: the common ADRC classes which have significantly increased after compression.

version shows signal loss after the compression. The masking effect in the noise perception [60] shows that the sensitivity of the human eye to signal distortion will decrease with local content activity. To generate the ground truth, it is then fair to decide that, if the loss is relatively large compared to the local content activity, it will be considered a coding artifact. The difference $d(i,j)$ at pixel position (i,j) between the uncompressed image X and the compressed version X_c is defined as:

$$d(i,j) = |X_c(i,j) - X(i,j)| \tag{2.2}$$

A threshold is defined as

$$Tc = kA(i,j) \tag{2.3}$$

where $A(i,j)$ is the local content activity in the corresponding DCT block of the uncompressed image and the activity is defined as the variance in the pixel values in the block. For the factor k, 0.1 is used since it generates results which match well with the perceived artifacts. If the difference $d(i,j) > Tc$, the pixel $X_c(i,j)$ is then considered to be a coding artifact pixel. Otherwise, it is not.

We test the artifact detector on the test material shown in Fig. 2.3. Table 2.1 shows the detection and false alarm rates for different test sequences. From the result, one can see that the detector gives a modest detection rate on average. Some image fragments from the ground truth and detection results of the sequence Bicycle are shown in Fig. 2.4. In the illustration, the ground truth artifact pixels and the correctly detected artifact pixels are marked by blue; the pixels that are not

(A) Bicycle   (B) Hotel   (C) Birds

(D) Lena   (E) Boat   (F) Motor
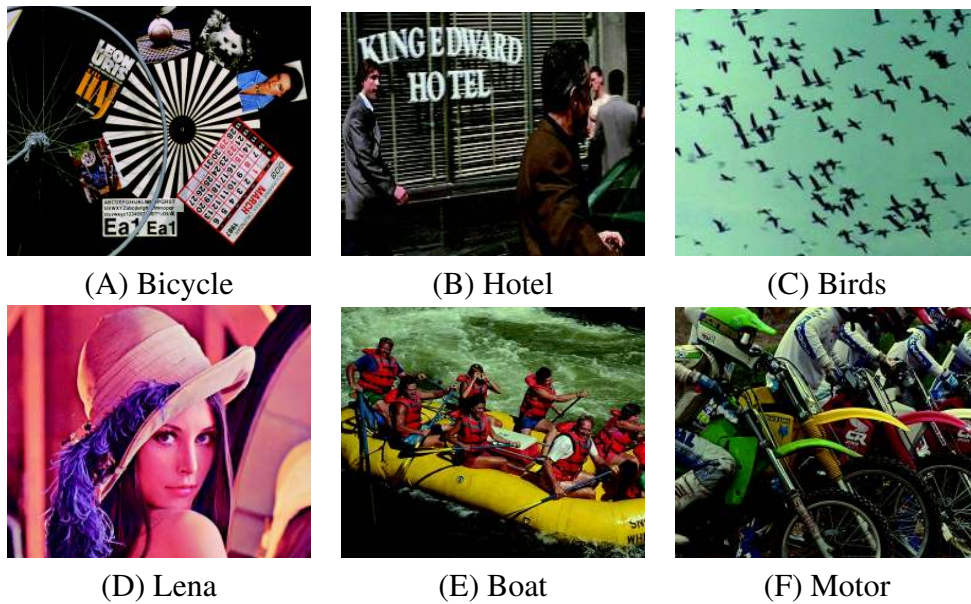
Figure 2.3: The testing material used for the evaluation.

artifacts and have been incorrectly detected are marked by red. Comparing the ground truth in Fig. 2.4 (C) and the detection result using these ADRC classes in Fig. 2.4 (D), one can see that most of blocking artifacts which are quite dominant in the image have been correctly detected. However, some ringing types of artifacts have not been detected. Since the ringing artifacts usually appear near strong edges, it is difficult to identify ringing artifacts with a limited filter aperture. Given the limitation, the detector gives a reasonable good detection.

One can also notice that the false alarm rate is quite high. In the result of using the ADRC classes in Fig. 2.4 (D), many real image edges have incorrectly been

Table 2.1: The detection and false alarm rates of using ADRC to detect artifacts.

Sequence	Detection rate	False alarm rate
Bicycle	65.2	10.6
Birds	78.3	12.2
Boat	68.9	21.5
Motor	50.3	16.6
Lena	64.8	19.5
Average	65.5	16.1

(A) uncompressed

(B) compressed

(C) ground truth

(D) ADRC detection result

(E) ADRC+DR detection result
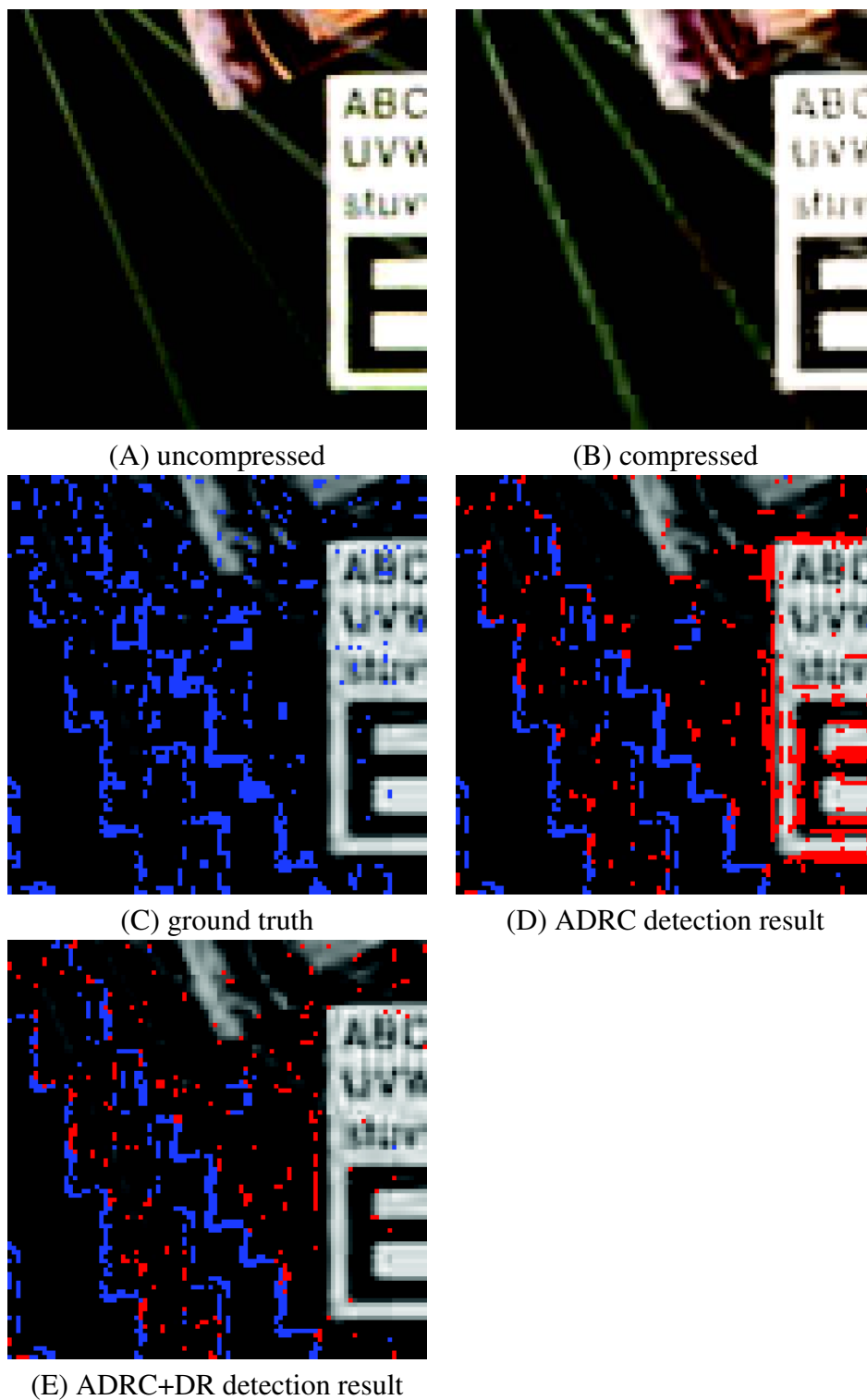
Figure 2.4: The artifact ground truth and the detection result: the ground truth artifact pixels and the correctly detected artifact pixels are marked by blue; the pixels that are not artifacts and have been incorrectly detected are marked by red.

detected as coding artifacts, most of which are horizontal and vertical edges, since these real image edges also have an identical ADRC pattern as these blocking artifact boundaries. The ADRC classification alone is clearly not enough to distinguish between the coding artifacts and the real image structures. This leads us to consider local contrast in the classification. From the literature it is known that the artifacts in low contrast area are more visible [59] according to human visual system. Local structure and contrast can usually be regarded as two orthogonal properties: local structure does not vary with local contrast, while local contrast does not depend on local structure. Clearly, image areas with edge patterns of different directions and high contrast are more likely to be the real image edges, while image areas with vertical or horizontal edge patterns and low contrast suggest possible coding artifacts. All of these suggest that one should combine local structure and contrast to detect coding artifacts.

To include local contrast in the classification, we calculate the histogram of the local contrast in the coding artifacts, which is shown in Fig. 2.5. The local contrast is defined as the difference between the maximum and minimal pixel value in the aperture. We can see that the coding artifacts are mainly distributed in the low contrast area.
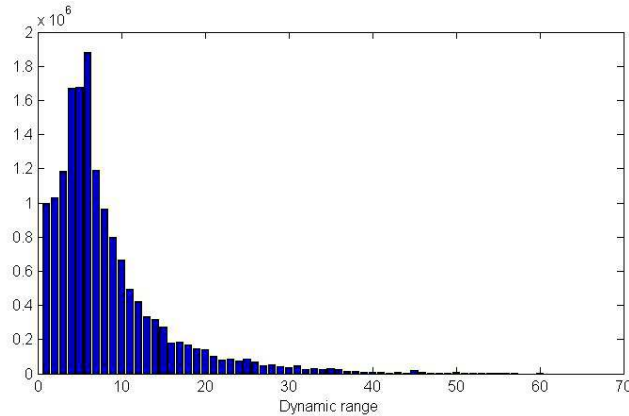
Figure 2.5: The histogram of the dynamic ranges in the coding artifacts.

Therefore, we add one extra bit, DR, to the ADRC code. The extra bit describes the contrast information in the aperture.

$$DR = \left\{ \begin{array}{ll} 0, & if \ x_{max} - x_{min} < Tr \\ 1, & otherwise \end{array} \right. \tag{2.4}$$

where Tr is the threshold value. The concatenation of $ADRC(x_i)$ of all pixels in the filter aperture and the extra bit DR gives the class code.
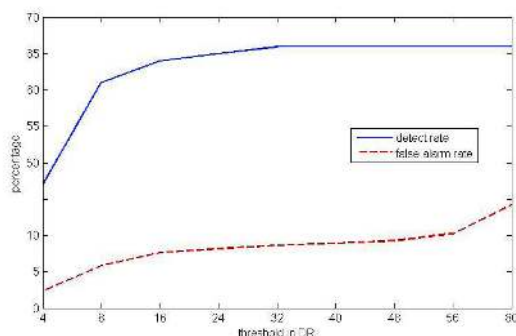
Figure 2.6: The plot of the detect and false alarm rate with the threshold used in the DR classification.

To find an optimal setting for the threshold Tr, we test different values for the threshold and plot them with the detection and false alarm rate using the mentioned ADRC classes and low contrast as the new detector in Fig. 2.6. As one can see, when the threshold is too low, the detection rate decreases since some of the artifacts in the low contrast area are not detected; the false alarm rate decreases because the number of detected pixels decreases. When the threshold is too high, the detection rate remains the same, but the false alarm will increase significantly. Overall, one can see that the threshold setting around 32 gives the best balance between the detect and false alarm rate. Fig. 2.4 (E) shows the detection results on the sequence Bicycle using the new detector. As one can see, the false alarms have been greatly reduced.

From the detection and false alarm rate result, one can see that the proposed detector does not give a perfect solution for coding artifact detection. This is due to the limited information in the filter aperture which is relatively small for determining whether it is a coding artifact pixel. Looking at a bigger scale would likely improve the detection performance, but that will also increase the cost. Nevertheless, the result has shown good indication that the combination of the ADRC and DR classification is effective at distinguishing coding artifacts. We expect that by including the ADRC and DR classification in the proposed processing framework, the resultant filter will have a probability weighted optimal processing, i.e., in the content classes that have a higher probability of being coding artifacts, the resultant filter will have a stronger artifact reduction effect.

2.3   Application I: Coding artifact reduction

Based on the coding artifact classification, we can apply the proposed content adaptive processing framework to remove the coding artifacts. As shown in the block diagram in Fig. 2.7, the process is similar to Kondo's concept except that the classification is done by the combination of ADRC and DR. We use a diamond shape aperture shown in the diagram to balance between performance and complexity.
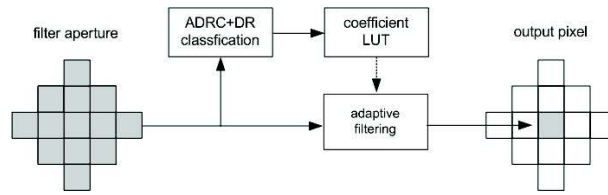
Figure 2.7: The block diagram of the proposed approach: the local image structure is classified using pattern classification and the filter coefficients are fetched from the LUT obtained from an off-line training.

The optimization procedure of the proposed method shown in Fig. 2.8 is also similar to Kondo's concept. To obtain the training set, we use original images as the reference output image. Furthermore, we compress the original images with the expected compression ratio. These corrupted versions of the original images are our simulated input images. The simulated input and the reference output pairs are classified using the same classification ADRC and DR on the input. Optimal coefficients are obtained by training the filters in individual classes.

In the following experiments, we will evaluate the proposed method in the applications of JPEG de-blocking and MPEG4-AVC/H.264 de-blocking. For JPEG de-blocking, we choose Nosratinia's method [63] (referred as *Nos*) as the comparison, since it is one of the methods which give the best results for JPEG de-blocking [58]. For MPEG4-AVC/H.264, we compare our proposed method with the in-loop filter used in the standard [62]. As an alternative method which applies in the spatial domain and does not require the block grid information, the bilateral filter [57] [100] (referred as *Bil*) is also included in the evaluation. The parameter settings for the bilateral filter are optimized for the compression level used in the experiments: the standard deviation of the Gaussian function for photometric similarity is set to 20 and the one for spatial closeness is set to 0.9. All the methods are optimized by using the same training set. The test images are shown in Fig. 2.3 and they are not included in the training set.

In order to enable a quantitative comparison, we first compress the original uncompressed test images using the same setting as in the training procedure. Then
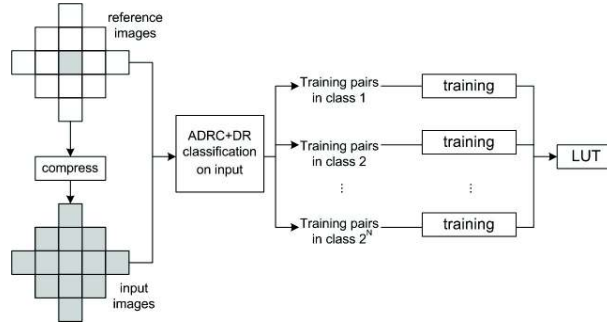
Figure 2.8: The training procedure of the proposed algorithm. The input and output pairs are collected from the training material and are classified by the mentioned classification method. The filter coefficients are optimized for specific classes.

we use the images as the simulated input images. The Mean Square Error (MSE) can be calculated from the original uncompressed images and the processed images.

2.3.1 JPEG de-blocking

We use the free baseline JPEG software from the Independent JPEG Group website [1] for JPEG encoding and decoding in the experiment. We apply the JPEG compression at the quality factor of 20 (the quality factor has a range from 1 to 100, and 100 is the best). Table 2.2 shows the MSE comparison of the evaluated methods. In term of the MSE score, one can see that the proposed method outperforms all the other methods, especially in the sequence Bicycle, which contains various image structures. The bilateral filter with the optimized parameter also achieves a similar result as Nosranatia's method. On average, the proposed method shows 25 percent improvement over the input.

To enable a qualitative comparison, image fragments from the image Motor processed by the mentioned methods are shown in Fig. 2.9. As one can see, the bilateral filter can reduce the coding artifacts significantly in flat areas, but it cannot suppress the artifacts in detailed areas. Nosranatia's method can remove the artifacts in the detailed areas, but it also loses some resolution because of the averaging. The proposed method shows the best result. It reconstructs the distorted details marked by a circle better than the bilateral filter because it adopts the image structure information. The processed image by the proposed method is the closest to the original.

---

[1]http://www.ijg.org/files/jpegscrc.v6b.tar.gz

Table 2.2: MSE scores for JPEG de-blocking.

Sequence	Mean Square Error			
	Input	*Nos*	*Bil*	Proposed
Bicycle	63.5	51.0	50.7	46.4
Birds	13.8	11.1	11.7	11.1
Boat	72.3	63.7	62.1	61.9
Motor	108.9	93.9	92.9	90.2
Lena	36.9	30.4	31.5	29.2
Average	59.1	50.0	49.8	47.8

2.3.2 H.264/MPEG4 AVC de-blocking

The H.264/MPEG4 AVC compression we used for evaluation is the reference software implementation of JM 11.0 system[2]. The in-loop filter is included in JM 11.0 system. The compression quality parameter QP, which has a range from 0 to 51, is set to 35. The MSE results for the sequence Hotel are shown in Fig. 2.10. We can see that the proposed method gives a better performance than the in-loop filter in every frame. And the advantage over the in-loop filter varies with the frames. The image fragments in Fig. 2.11 show that although the in-loop filter can reduce most of the blocking artifacts significantly in the flat area due to its in-loop advantage, it does not repair the corrupted edge structures marked by a circle well which can be nicely reconstructed with the proposed post-processing method.

2.4 Application II: Resolution up-conversion integration

Although HDTV has become a standard appliance in every household today, there are still a lot of legacy videos in standard definition. Additionally they are compressed using different compression standards. The standard definition content has to be up-scaled to fit the resolution. However, coding artifacts will be preserved and enlarged after the resolution conversion. These coding artifacts will be even more difficult to remove.

Resolution up-conversion is traditionally approached by linear methods, such as bi-linear and bi-cubic interpolations, which usually blur image details. Advanced resolution up-conversion algorithms [98][105][102] have been proposed

---

[2]http://iphome.hhi.de/suehring/tml/download/
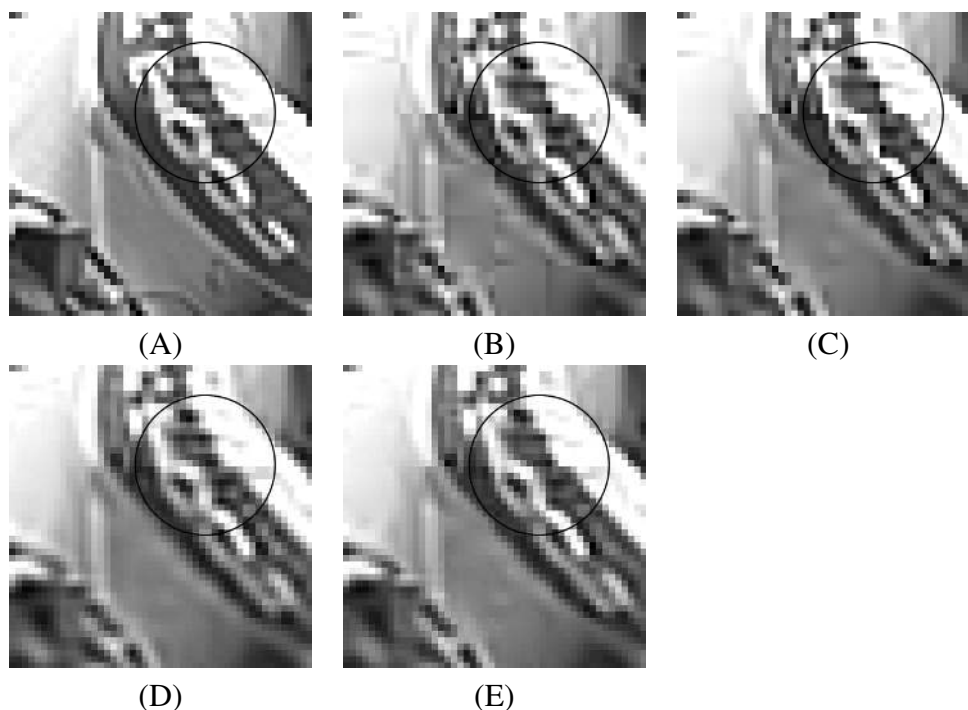
Figure 2.9: Image fragments from the image Motor: (A) uncompressed original, (B) JPEG compressed input, (C) processed by the bilateral filter, (D) processed by Nosratinia's method, (E) processed by the proposed method.

to be adaptive to local structure or edge orientation, which makes them capable of preserving edges and fine details in the image content. Zhao compared the state-of-the-art up-scaling techniques both objectively and subjectively, and concluded that the structure-adaptive LMS training technique, proposed by Kondo, performed the best [80]. However, Kondo's method was designed for images with little noise. When the image is compressed, coding artifacts will be preserved and enlarged after up-scaling. These coding artifacts, e.g. blocking artifacts, will be even more difficult to remove than those in the original low resolution image, because the coding artifacts will spread among more pixels and become not trivial to detect after the resolution up-conversion. In the video processing chain, coding artifacts are usually suppressed using certain low-pass filtering before applying resolution up-conversion. Therefore, image details can be blurred by the low-pass filtering and cannot be recovered during the resolution up-conversion.

We propose an integrated artifact reduction and resolution up-conversion approach using the proposed framework in this section. Based on the proposed coding artifact classification, optimal LMS filters are used for estimating the high resolution pixels. Since the classification can distinguish between coding arti-
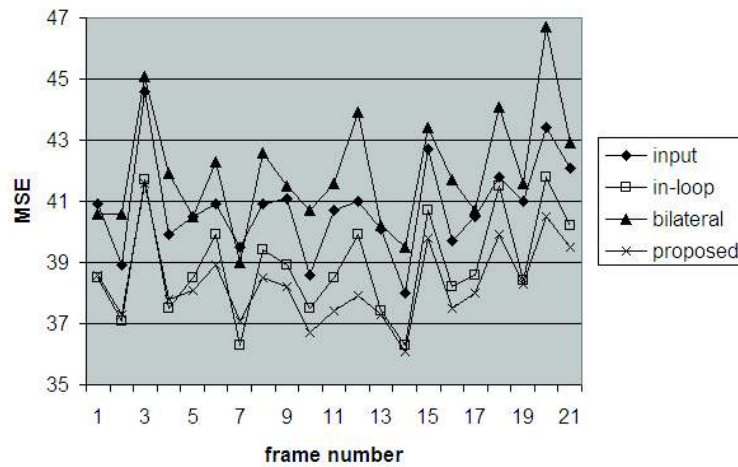
Figure 2.10: The MSE results of H.264 de-blocking for Hotel sequence.

facts and real image structures, we could expect that the proposed method will reduce coding artifacts. Because the classification also includes the structure information, we hope that different image structures and fine details will be well preserved. The optimal coefficients are obtained by a training between the high resolution reference and the simulated degraded version.

The optimization procedure of the proposed method is shown in Fig. 2.12. To obtain the training set for combined up-scaling and artifact reduction, we use original images as the reference output image and the down-scaled and compressed version as the simulated input. To get the down-scaled and compressed version, the original images are first down-sampled using a bi-linear filter. The down-sampled images are then compressed to introduce coding artifacts.

For the evaluation, the proposed algorithm is benchmarked against two alternative solutions generated by cascading the resolution up-conversion method proposed by Kondo [97] and the artifact reduction method proposed by Zhao [80] in different orders. These two methods have significant advantages over other heuristically designed filtering techniques. For a fair comparison, an ADRC code of a 3x3 aperture is used for classification in the up-scaling. The classification method used in coding artifact reduction proposed by Zhao is the combination of the ADRC and the relative position of a pixel in the coding block grid. A diamond shape aperture with 13 pixels is used, which requires 12 bits for ADRC and 4 bits for relative position coding. The drawback of this method is that block grid positions are not always available, especially for scaled material. For the cascaded method of first applying resolution up-conversion then doing coding artifact reduction, the classification of coding artifact reduction is carried out on the
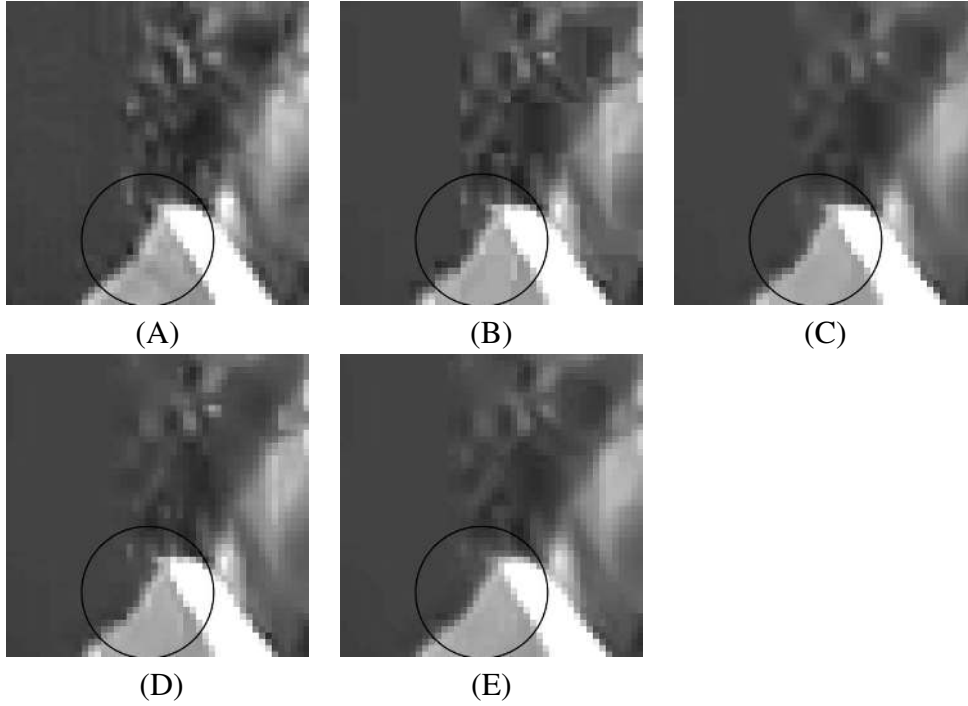
Figure 2.11: Image fragments from the sequence Hotel: (A) original uncompressed, (B) H.264 compressed, (C) processed by bilateral filter, (D) output of the in-loop filter, (E) processed by the proposed method.

up-scaled HD signal and the relative position of a pixel in the block grid is also up-scaled accordingly to suit the HD signal. The coefficients of both methods are obtained by the LMS technique.

For the cost comparison, Table 2.3 shows the numbers of coefficients that need to be stored in look-up tables (LUT) and the numbers of additions and multiplications needed for outputting every high resolution pixel for each of these three algorithms. The proposed algorithm is much more economical than the other two in terms of LUT size. Since the training process is done offline and only needs to be done once, thus the computational cost is limited for all the three methods.

We test the algorithms on a variety of test sequences, which are first down-sampled then compressed using the same setting as during the training. Fig. 2.13 shows the image fragments of these test sequences we used for the experiment. All the test sequences are excluded from the training set. The objective metric we use is mean square error (MSE), i.e. we calculate the MSE between the original HD sequences and the result sequences processed on the compressed down-scaled versions of the original sequences.

Table 2.4 shows the results of the proposed algorithm in comparison to the re-
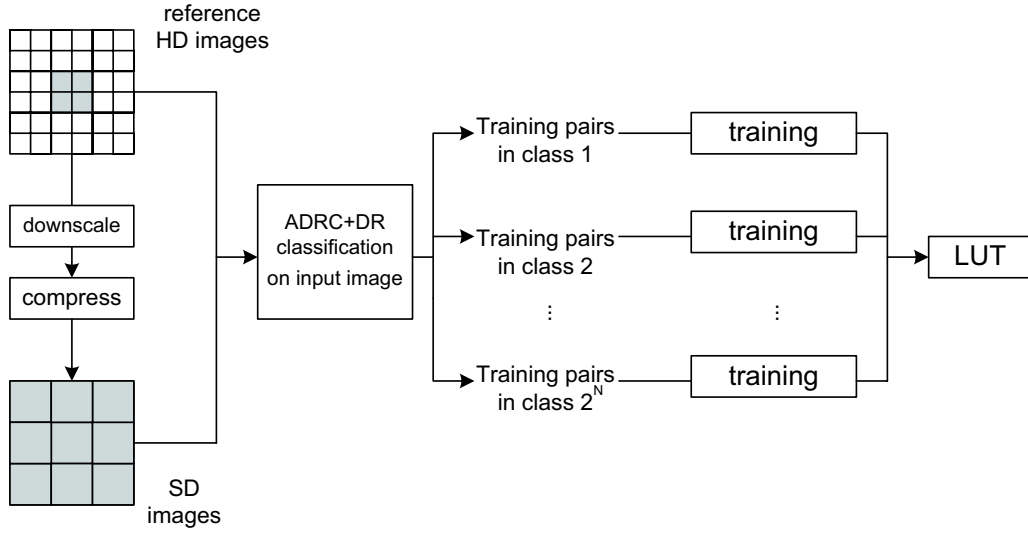
Figure 2.12: The training procedure of the proposed method for combined up-scaling and arti-fact reduction.

Table 2.3: Cost comparison for the proposed integration and alternatives.

Cost	Computational cost by algorithms		
	Zhao+Kondo	Kondo+Zhao	Proposed
Coefficients	4096x16x13 + 256x9	256x9 +4096x16x13	256x2x9
Multiplications	13/4+9	9+13	9
Additions	13/4+9	9+13	9

sults of first applying coding artifact reduction then up-conversion and first apply-ing up-conversion then artifact reduction. The result of resolution up-conversion using Kondo's method without applying artifact reduction is also shown for refer-ence. From the results, one can see that the proposed algorithm outperforms the other two concatenated methods for all sequences. The results also reveal that the order of applying up-conversion and artifact reduction affects the performance of the concatenated method. For some sequences, applying artifact reduction first gives better results; for other sequences, vice versa.

For a qualitative comparison, Fig. 2.14 shows fragments from the Bicycle sequence processed by all the three methods. As one can see, the result of first ap-plying up-conversion then artifact reduction contains more residual artifacts than the proposed algorithm, because up-scaling makes coding artifacts spread out in more pixels and more difficult to remove. The result of first applying artifact

(A) Bicycle   (B) Hotel

(C) Parrot   (D) Teeny

Figure 2.13: The testing material used for the evaluation.

Table 2.4: MSE scores for different methods.

Sequence	MSE			
	Kondo	Zhao+Kondo	Kondo+Zhao	Proposed
Hotel	116.3	113.4	108.5	102.4
Parrot	36.1	32.2	35.1	31.7
Teeny	66.9	59.9	63.7	59.0
Bicycle	183.5	164.3	170.2	163.1
Average	100.7	92.5	94.4	89.0

(A)

(B)

(C)

(D)
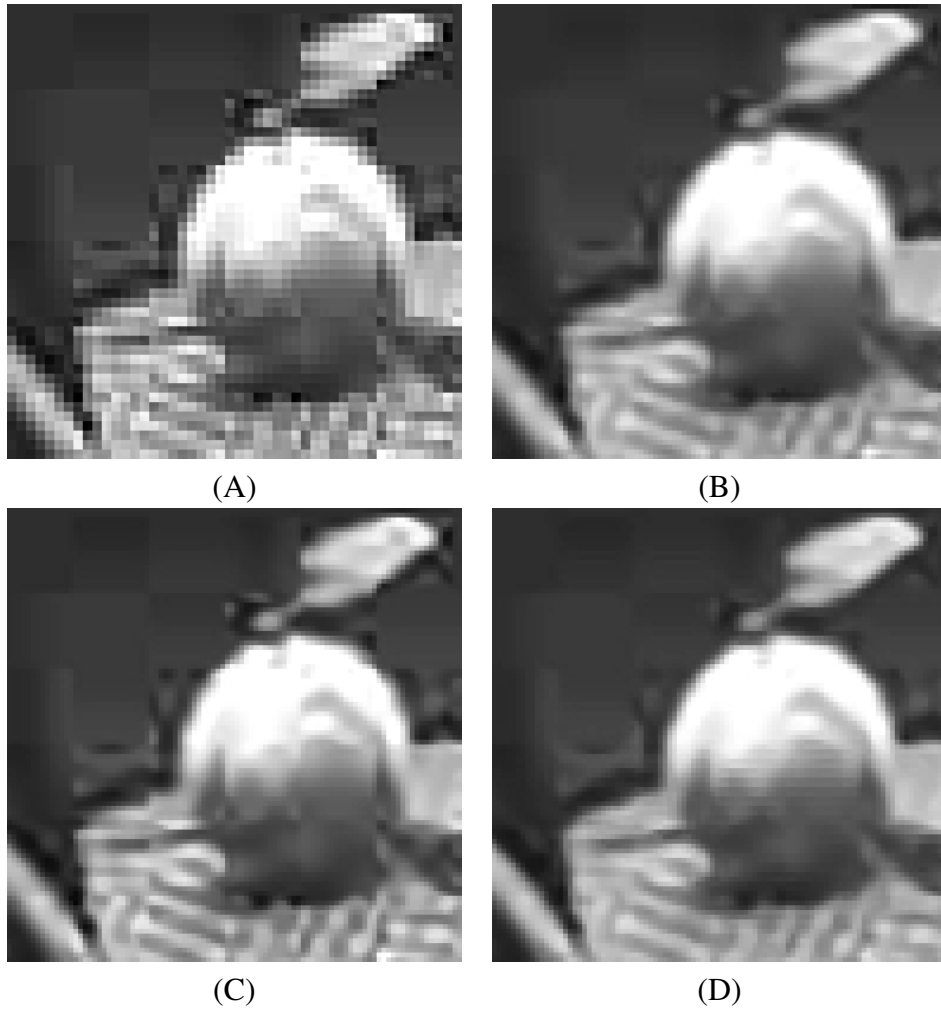
Figure 2.14: Image fragments from the sequence Bicycle: (A) simulated input, (B) processed by the proposed method, (C) processed by Kondo + Zhao, (D) processed by Zhao + Kondo.

reduction then resolution up-conversion is blurrier than our proposed algorithm, because the artifact reduction step blurs some details, which cannot be recovered by the up-scaling step.

2.5 Application III: Sharpness enhancement integration

Sharpness enhancement is another essential video enhancement usually included in the video processing chain. To increase the perceptual quality of compressed image, a sharpness enhancement algorithm can be applied after the artifact reduction process. However one separate step will take more time and more memory, and will eventually increase the cost of the whole process. Here, we propose an integrated approach combining coding artifact reduction and sharpness enhancement.

To integrate with sharpness enhancement, one can train the filter to learn the process to turn a compressed version of reference images into a sharpen version of reference images. However, it is not clear how to obtain the optimal sharpness enhancement. Therefore, we propose to train the filter to turn a blurred and then compressed version of reference images to the original reference ones. We expect that the resulting filter will have the behavior of inverting the process of blur and compression, that is, the integrated sharpness enhancement and artifact reduction. An isotropic Gaussian function is used to blur the reference images. Then better sharpness enhancement can be obtained using the prior classification information. For example, the edge will be sharpened across the direction of edge and the coding artifacts will not be enhanced. The optimization procedure is shown in Fig. 2.15. To obtain the training set, we blur the original images with an isotropic Gaussian blur kernel and compress them with the expected compression ratio. These blurred and corrupted versions of the original images are our simulated input images. One can adjust the Gaussian blur radius to change the degree of sharpness enhancement. The process of applying the proposed approach is similar to the diagram shown in Fig. 2.7.

In order to obtain a subjective evaluation of the proposed method, a paired comparison of compressed test sequences and their post-processed versions was performed. The test set with CCIR-601 resolution includes five stills compressed using JPEG at a quality factor of 20, and six video sequences compressed using MPEG2 at a bit-rate of 2.5Mbit/s. Every test material and its post-processed version were shown next to each other on an LCD screen in a randomized order. Eighteen expert and non-expert viewers were gathered to do the paired comparison one by one. Each of them was asked to sit in front of the screen at a distance
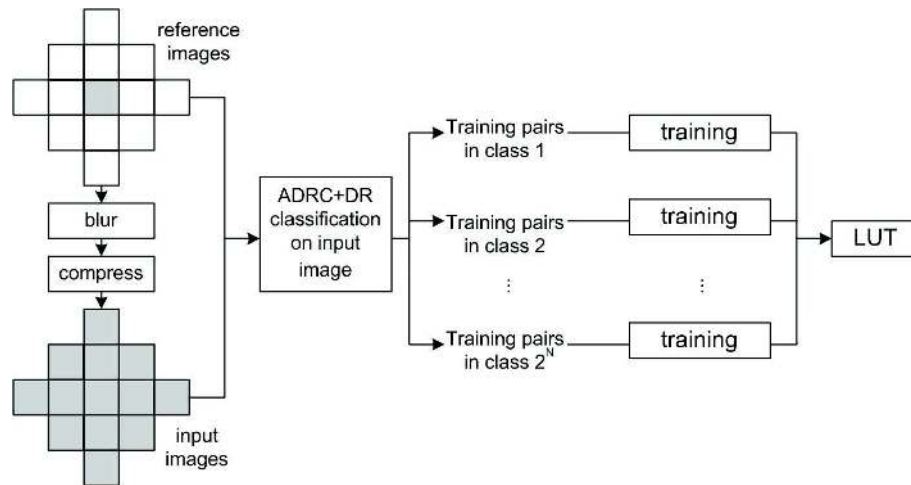
Figure 2.15: The training procedure of the proposed algorithm. The input and output pairs are collected from the training material and are classified by the mentioned classification method. The filter coefficients are optimized for specific classes.

of six times the screen height and select the one that he/she perceived as having the better image quality.

The evaluation result shows that the post-processed version was chosen 329 times against 67 times for the original. An analysis of the result as proposed by Montag [5] shows that the average image quality scale of the original has a 95 percent confidence interval (CI) of 0 ± 0.067 as the reference, and the 95 percent CI of the image quality scale for the proposed method is 0.96 ± 0.067. This suggests that the perceptual image quality has been significantly increased by the proposed method. Image fragments from a compressed sequence and the post-processed version by the proposed method are also illustrated in Fig. 2.16. As one can see, the proposed method has great effectiveness at removing the artifacts while enhancing the sharpness.

2.6 Conclusion

In this chapter, we have designed a new classifier suitable to distinguish coding artifacts from image details. The classifier classifies the local image content using two orthogonal properties, local structure and local contrast. Incorporating the new classifier for coding artifact detection into the proposed content adaptive filtering framework leads to various video enhancement algorithms for digitally coded videos, including artifact reduction and its integrations with sharpness and resolution enhancement. These algorithms not only show better performance at

(A) Compressed

(B) Processed

Figure 2.16: Image fragments from the compressed sequence Hotel (A) and the post-processed by our proposed method (B).

picture quality, but also feature low cost in terms of hardware implementation.

The application of combined sharpness enhancement and coding artifact reduction shows that it is possible to combine two opposite operations, sharpening and smoothing, at the same time, with the help of the coding artifact classifier.

From the analysis of the proposed classification for coding artifact detection, we see that there is a limitation of using local content information only from the filter aperture. Looking at the local content at a bigger scale, for instance, using local statistics of the content classes, is expected to lead to more improvements to the detection. Such a topic is interesting for future research.

In the following chapters, we will continue to explore new classifiers for the content classification part in the proposed framework.

Chapter 3

Content classification in blurred videos

In the previous chapter, we have introduced a coding artifact classifier to the content classification part of the proposed content adaptive processing framework. Besides the coding artifacts in digital videos, there is another type of signal degradation, focal blur, or out-of-focus blur, which has also received considerable attention in the field of video enhancement.

Focal blur in images and videos occurs when objects in the scene are placed outside the focal plane of the camera. Due to a limited focal range of optical lenses, or sub-optimal settings of the camera, objects may suffer from blur degradation in the registered image.

Moreover, as objects may have varied distances to the lens, they are often differently blurred in the registered image. We therefore conclude that an, accurate, local blur estimator would promise interesting additional applications in the video enhancement domain of this thesis.

For example, one could think of applying local blur estimation to restore differently blurred parts of an image, resulting in an all-in-focus result. We further expect that blocking artifacts are most visible in the out-of-focus areas. Hence, knowledge of the local blur could also be beneficial in a blur-adaptive coding artifact reduction application.

Many local blur estimation methods have been proposed to estimate the spatially variant blur. However, they are typically based on low-level image clues and therefore can, neither generate consistent blur estimation over objects, nor distinguish focal blur from other non-degradation blur, for example, shading blur.

In this chapter, we propose a new local blur estimation method that generates consistent blur estimates for objects in an image. First, a novel local blur estimator based on edges is introduced. It uses a Gaussian isotropic point spread function model and the maximum of difference ratio between the original image and its

two digitally re-blurred versions is calculated to estimate the local blur radius. The advantage over alternative local blur estimation methods is that it does not require edge detection, has a lower complexity and does not degrade when multiple edges are close.

With the blur estimates from the proposed blur estimator and other clues from the image, like color and spatial position, the image is segmented using clustering techniques. Then within every segment, the blur radius of the segment is estimated to generate a blur map that is consistent over objects.

Finally, we apply the novel segmentation-based blur estimator into the proposed framework, which has resulted in solutions for the two mentioned applications, all-in-focus image restoration and blur-adaptive coding artifact reduction.

The rest of this chapter is organized as follows. In Section 1, we present the proposed blur estimation algorithm and its analysis based on an ideal edge model and we compare it with the most relevant alternative, Elder's method. Section 2 shows the proposed segmentation-based blur estimation. Two applications, all-in-focus image restoration and blur dependent enhancement, and their results are presented in Section 3. Finally, we conclude the chapter in Section 4.

3.1   Introduction

Focal blur often occurs in images or videos due to finite depth-of-field. There are also other types of blur in images, penumbral blur and shading blur, which have a similar appearance as focal blur [74]. Fig. 3.1 illustrates how these three types of blur are formed. Penumbral blur is caused by a shadow that exhibits a penumbra when the light source is not a point source. Shading blur is generated by the smooth curved surface of an illuminated object.
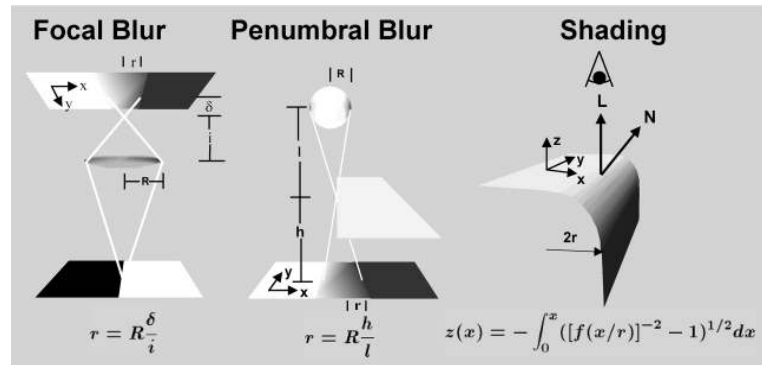
Figure 3.1: Three types of blur: focal blur due to finite depth-of-field; penumbral blur at the edge of a shadow; shading blur at a smoothed object edge. Source [74].

These types of blur are usually modeled as Gaussian blurring [53]. Therefore, the problem of blur estimation is to identify the Gaussian point spread function (PSF). Many techniques [72] [73] have been proposed to estimate the point spread function of the spatially invariant blur. For local blur estimation, methods are typically based on an analysis of an ideal edge signal. In Elder's method [74] the blurred edge signal is convolved with the second derivative of the Gaussian function and the response has a positive and a negative peak. The distance between these peak positions can be used to determine the blur radius. Another approach from Kim [78] is based on an isotropic discrete point spread function (PSF) model. The one-dimensional step response along the orthogonal direction of edge direction will be estimated and the PSF can be obtained by solving a set of linear equations related to the step response. Both Elder's and Kim's method require detection of the edge direction, which adds complexity to the algorithm.

3.2 Local blur estimation

We propose a new blur estimation method based on the difference between two digitally re-blurred versions of an image. The insight, on which the proposed method is based, starts from the observation that the Fourier analysis of a perfect edge shows a fixed ratio of the energy in different frequency bands. Focal blur will change the ratio by weakening the higher frequencies. By measuring the ratio in the frequency bands, an estimate of the focal blur can be obtained. The two re-blurred versions correspond to the energy in two frequency bands. That means that the two re-blurring operations are arranged to extract mutually different portions of the spatial spectrum of the input image. If the ratio between the energy in these two frequency bands is relatively high, then the value of the blur measure is also relatively high. Here, through an analysis performed on an edge model, we show that the blur radius can be easily calculated from the difference ratio, independent from the edge amplitude or position.

First, we analyze the blur estimation with a one dimensional (1D) signal. We assume an ideal edge signal and a discrete Gaussian blur kernel. The edge is modeled as a step function with amplitude A and offset B. For a discrete signal, the edge $f(x)$ shown in Fig. 3.2 is

$$f(x) = \begin{cases} A + B, & x \geq 0 \\ B, & x < 0 \end{cases}, x \in \mathbb{Z} \tag{3.1}$$

where x is the position. The focal blur kernel is modeled by a discrete Gaussian function:

$$g(n, \sigma) = C(\sigma) \exp\left(-\frac{n^2}{2\sigma^2}\right), n \in \mathbb{Z} \tag{3.2}$$
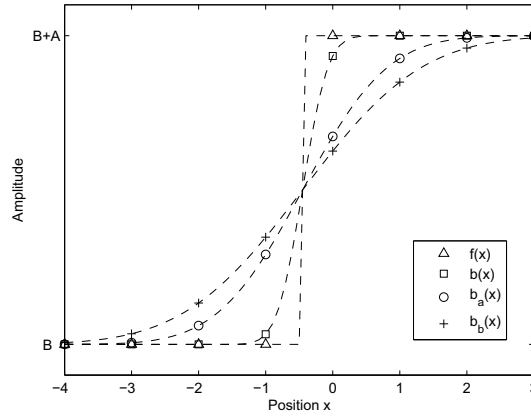
Figure 3.2: The step edge $f(x)$, the blurred edge $b(x)$ and its two re-blurred versions $b_a(x)$, $b_b(x)$

where σ is the unknown blur radius to be estimated and $C(\sigma)$ is the normalization factor. The normalization implies:

$$\sum_{n \in \mathbb{Z}} g(n, \sigma) = \sum_{n \in \mathbb{Z}} C(\sigma) \exp\left(-\frac{n^2}{2\sigma^2}\right) = 1 \tag{3.3}$$

$C(\sigma)$ admits no closed form expression, but the approximation $\frac{1}{\sqrt{2\pi}\sigma}$ can be considered acceptable when $\sigma > 0.5$. Then the blurred edge $b(x)$ will be:

$$
\begin{aligned}
b(x) &= \sum_{n \in \mathbb{Z}} f(x - n) g(n, \sigma) \\
&= \begin{cases} \frac{A}{2}\left(1 + \displaystyle\sum_{n=-x}^{x} g(n, \sigma)\right) + B, & x \geq 0 \\ \frac{A}{2}\left(1 - \displaystyle\sum_{n=x+1}^{-x-1} g(n, \sigma)\right) + B, & x < 0 \end{cases}, x \in \mathbb{Z}
\end{aligned}
\tag{3.4}
$$

As the convolution of two Gaussian functions with blur radii σ_1, σ_2 is:

$$g(n, \sigma_1) * g(n, \sigma_2) = g(n, \sqrt{\sigma_1^2 + \sigma_2^2}) \tag{3.5}$$

Re-blurring the blurred edge using Gaussian blur kernels with blur radius σ_a and

σ_b ($\sigma_b > \sigma_a$), results in two re-blurred versions $b_a(x)$ and $b_b(x)$:

$$b_a(x) = \begin{cases} \frac{A}{2}\left(1 + \sum_{n=-x}^{x} g(n, \sqrt{\sigma^2 + \sigma_a^2})\right) + B, & x \geq 0 \\ \frac{A}{2}\left(1 - \sum_{n=x+1}^{-x-1} g(n, \sqrt{\sigma^2 + \sigma_a^2})\right) + B, & x < 0 \end{cases} \quad , x \in \mathbb{Z} \qquad (3.6)$$

$$b_b(x) = \begin{cases} \frac{A}{2}\left(1 + \sum_{n=-x}^{x} g(n, \sqrt{\sigma^2 + \sigma_b^2})\right) + B, & x \geq 0 \\ \frac{A}{2}\left(1 - \sum_{n=x+1}^{-x-1} g(n, \sqrt{\sigma^2 + \sigma_b^2})\right) + B, & x < 0 \end{cases} \quad , x \in \mathbb{Z} \qquad (3.7)$$

To make the blur estimation independent of the amplitude and offset of edges, we calculate the ratio $r(x)$ of the differences between the original blurred edge and the two re-blurred versions for every position x:

$$r(x) = \frac{b(x) - b_a(x)}{b_a(x) - b_b(x)}$$

$$= \begin{cases} \dfrac{\sum_{n=-x}^{x} \left(g(n, \sqrt{\sigma^2 + \sigma_a^2}) - g(n, \sigma)\right)}{\sum_{n=-x}^{x} \left(g(n, \sqrt{\sigma^2 + \sigma_b^2}) - g(n, \sqrt{\sigma^2 + \sigma_a^2})\right)}, & x \geq 0 \\[4ex] \dfrac{\sum_{n=x+1}^{-x-1} \left(g(n, \sqrt{\sigma^2 + \sigma_a^2}) - g(n, \sigma)\right)}{\sum_{n=x+1}^{-x-1} \left(g(n, \sqrt{\sigma^2 + \sigma_b^2}) - g(n, \sqrt{\sigma^2 + \sigma_a^2})\right)}, & x < 0 \end{cases} \qquad (3.8)$$

The difference ratio peaks at the edge position $x = -1$ and $x = 0$ as shown in Fig. 3.3. So we obtain:

$$r(x)_{max} = r(-1) = r(0) = \frac{\frac{1}{\sigma} - \frac{1}{\sqrt{\sigma^2 + \sigma_a^2}}}{\frac{1}{\sqrt{\sigma^2 + \sigma_a^2}} - \frac{1}{\sqrt{\sigma^2 + \sigma_b^2}}} \qquad (3.9)$$

When σ_a, $\sigma_b \gg \sigma$, we can use some approximations:

$$\sqrt{\sigma^2 + \sigma_a^2} \approx \sigma_a$$
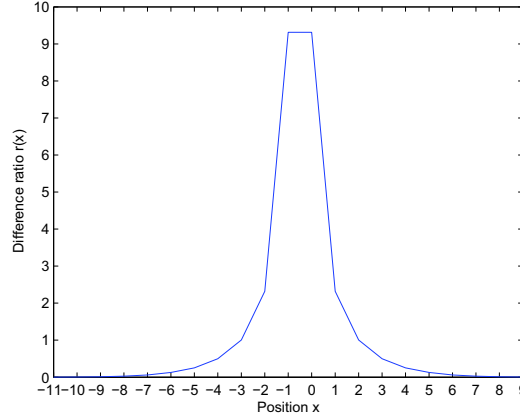
Figure 3.3: the difference ratio plot among the edge.

$$\sqrt{\sigma^2 + \sigma_b^2} \approx \sigma_b$$

which we use to simplify Equation 3.9:

$$r(x)_{max} \approx \frac{\frac{1}{\sigma} - \frac{1}{\sigma_a}}{\frac{1}{\sigma_a} - \frac{1}{\sigma_b}} = \frac{\left(\frac{\sigma_a}{\sigma} - 1\right) \cdot \sigma_b}{\sigma_b - \sigma_a} \tag{3.10}$$

or

$$\sigma \approx \frac{\sigma_a \cdot \sigma_b}{(\sigma_b - \sigma_a) \cdot r(x)_{max} + \sigma_b} \tag{3.11}$$

Equation 3.9 and 3.11 show that blur radius σ can be calculated from the difference ratio maximum $r(x)_{max}$ and the re-blur radii σ_a, σ_b, *independent* of the edge amplitude A and offset B. The identification of the local maximum of difference ratio $r(x)_{max}$ not only estimates the blur radius, but also locates the edge position, which implies the blur estimation does not require a separate edge detection.

For the blur estimation in *images*, i.e. two dimensional (2D) signals, we use a 2D isotropic Gaussian blur kernel for the re-blurring. As any direction of an isotropic Gaussian function is a 1D Gaussian function, the proposed blur estimation is also applicable. Using 2D Gaussian kernels for the estimation avoids detecting the angle of the edge or gradient, as required in Elder's and Kim's method. This helps to keep the complexity low.

In the simplest version, we implement the algorithm in a block-based manner to obtain a blur map on a block-grid of a natural image. As shown in the block diagram in Fig. 3.4, the difference ratios are calculated pixel-wise using the original image and its two re-blurred versions. Then, in every block the maximum of

the difference ratio is used to determine the blur radius in the block. A block size of 8×8 pixels has been used. Finally we assign the estimated blur radius to all pixels within the block.
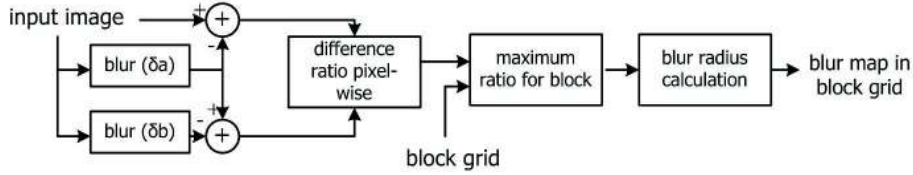
Figure 3.4: The block diagram of the proposed block based blur estimation.

For the evaluation, we test the proposed method on some synthetic and natural images, and we compare that with Elder's methods since this is considered state-of-the-art [75]. For the synthetic images, we use multiple step edges blurred by a 1D Gaussian blur kernel, with the blur radius increasing linearly along the edge from 0.1 to 5, as shown in Fig. 3.5. About one percent Gaussian noise is added to simulate sensor noise. Different distances between neighboring step edges D have been used.

D=50 pixels   D=20 pixels
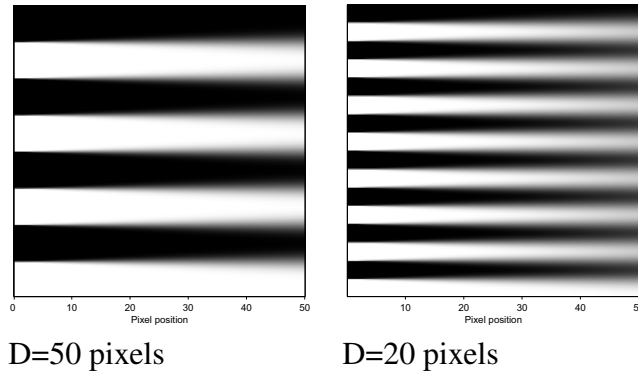
Figure 3.5: Synthetic images used in the test of the blur estimation algorithms.

We use the optimal settings for both methods and the results are shown in Fig. 3.6. As one can see, when the distance between neighboring edges is relatively large ($D = 50$ pixels) both methods can reliably estimate a wide range of blur radii. When the distance between neighboring edges becomes relatively small ($D = 20$ pixels), Elder's method suffers considerably from the interference of neighboring edges and the estimation is very unreliable while the propose method demonstrates clearly better estimates.

D=50 pixels   D=20 pixels
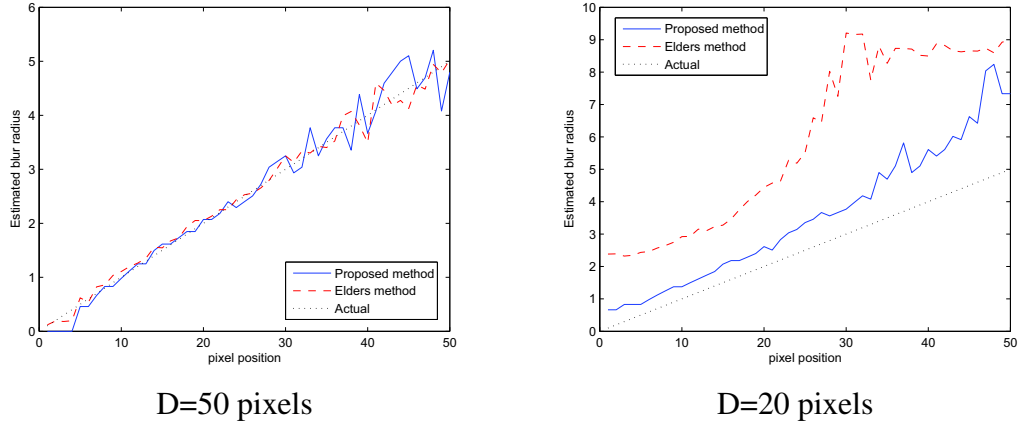
Figure 3.6: Estimated blur radius on the synthetic images for Elder's method and the proposed method.

For natural images, we use the image Lena for the test. This image is shot with a soft-focus technique and objects in this image have been differently blurred. Blur estimation results of Elder's method and the proposed method have been illustrated in grey images (Fig. 3.7). Both methods are implemented in a block-based manner. In the blur estimation results, the darker areas indicate a larger blur radius, while the lighter areas indicate a smaller blur radius. We can see that the differently blurred background of the image Lena has been estimated more accurately by the proposed method than by Elder's method. Note that Lena's face and shoulder show strong *shading blur*. Both methods can estimate the blur radius but they cannot distinguish the shading blur from the focal blur.

3.3   Object blur estimation

As shown in the previous section, the local blur estimation is based on the edge signal in the image. This means that reliable blur estimation only exists at the edges. From the results in Fig. 3.7, we see that the generated blur map is very inconsistent. It is often desirable to obtain a consistent blur estimation over objects to allow more stable restoration or enhancement. Another weakness of the proposed method is that the smooth transition on the surface of in-focus objects, such as shading blur, is typically estimated to have a large blur radius although there is no degradation. To alleviate these problems, we propose three approaches to improve the consistence of the blur map. We shall first describe the three proposals in the following subsections and then analyze their performance.
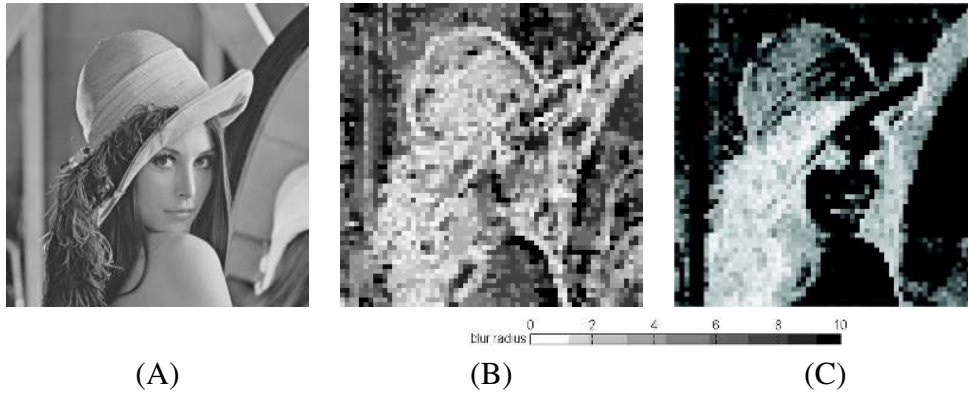
Figure 3.7: Estimation results: the darker areas indicate a larger blur radius, while the lighter areas indicate a smaller blur radius. (A) input image Lena, (B) result of Elder's method, (C) result of the proposed method.

3.3.1 Spatial-temporal neighborhood approach

A solution to an analogous problem can be found in the literature on motion estimation [71]. In the paper, reliable motion estimates are available only at certain detailed places, but not in homogeneous areas and along the edge direction. That problem is quite similar to the problem described here. In [71] an image is scanned in a block-based manner. It is assumed that objects are bigger than blocks and that there is a high probability that the motion vector of an unknown block is the same as one of the neighboring blocks. The neighboring blocks that have been updated in the current iteration are called spatial neighbors and the others are called temporal neighbors, since they are available from the *previous* frame or the *previous* iteration.

A similar approach [70] can be attempted to estimate the blur in low contrast areas. After the blur has been estimated for every location, it is proposed to scan through the image, updating the blur estimate of the processed block from a set of candidates. The candidates are shown in Fig. 3.8 if the image is scanned from top-left to bottom-right, where blocks marked by S are considered spatial neighbors (from the current iteration) and blocks marked by T are considered temporal neighbors (from the previous iteration or frame).

The block in the center is the one to be processed and the blur values from this block and its neighboring blocks together form the candidate set. The processed block is updated with the value from the candidate which has the best similarity with the processed block. This similarity can be expressed by means of weighting factors. For the weights, we propose to use the luminance amplitude of a current block and the difference in average chrominance between the processed

S	S	S
S	T	T
T	T	T

Figure 3.8: Spatial-temporal neighborhood.

block and the neighboring block. Since the most reliable estimates are available at edge positions, the amplitude of a block should be a good indication for the reliability of that block. Additionally, it is assumed that neighboring blocks from the same object have the similar chrominance values and that neighboring blocks from different areas have different chrominance values.

Let $A(bi, bj)$ denote the amplitude of the luminance signal in a block, where bi, bj are the horizontal and vertical position of the block, respectively. For the amplitude, we use the difference between the maximum and minimum luminance value of all the pixels in the block. Let $U(bi, bj)$ and $V(bi, bj)$ denote the average chrominance values in the block. Furthermore, let $\{U(bi + bk, bj + bl), V(bi + bk, bj + bl) : bk, bl = -1, 0, 1.\}$ be the chrominance values in the neighboring blocks. The chrominance similarity weight $UV(bk, bl)$ is defined as:

$$UV(bk, bl) = 1 - \frac{|U(bi + bk, bj + bl) - U(bi, bj)|}{U_{max} - U_{min}} \cdot \frac{|V(bi + bk, bj + bl) - V(bi, bj)|}{V_{max} - V_{min}}$$
(3.12)

where $U_{max}, U_{min}, V_{max}, V_{min}$ are the maximum and minimum chrominance value in the current frame, respectively. As we can see, the similarity weight has a range between 0 and 1. A value close to 1 indicates a high similarity and a value close to 0 indicates a low similarity.

Let $\sigma(bi, bj)^t$ denote the blur estimate in the block at iteration t. Then the candidates of the blur estimates are defined as:

$$C(\vec{D}) = \begin{pmatrix} \sigma(bi - 1, bj - 1)^t & \sigma(bi - 1, bj)^t & \sigma(bi - 1, bj + 1)^t \\ \sigma(bi, bj - 1)^t & \sigma(bi, bj)^{t-1} & \sigma(bi, bj + 1)^{t-1} \\ \sigma(bi + 1, bj - 1)^{t-1} & \sigma(bi + 1, bj)^{t-1} & \sigma(bi + 1, bj + 1)^{t-1} \end{pmatrix}$$
(3.13)

The vector $\vec{D}$ indicates which value is given to the processed block. This vector is calculated from the candidate set CS:

$$CS = \begin{pmatrix} UV(-1, -1)A(-1, -1)^t & UV(-1, 0)A(-1, 0)^t & UV(-1, 1)A(-1, 1)^t \\ UV(0, -1)A(0, -1)^t & UV(0, 0)A(0, 0)^{t-1} & UV(0, 1)A(0, 1)^{t-1} \\ UV(1, -1)A(1, -1)^{t-1} & UV(1, 0)A(1, 0)^{t-1} & UV(1, 1)A(1, 1)^{t-1} \end{pmatrix}$$
(3.14)

The vector $\vec{D}$ is chosen with respect to the maximum in the candidate set:

$$\vec{D} = \arg \max_{(bk,bl)} CS \tag{3.15}$$

And the blur estimate of the processed block is updated according to the candidate pointed by the displacement vector:

$$\sigma(bi, bj)^t = C(\vec{D}) \tag{3.16}$$

The amplitude of the processed block is also updated with the amplitude of the chosen candidate block. Because the reliability of the block decreases with the distance from the original block a weighting factor is used to decrease the reliability of this block. The updated amplitude is then set by:

$$A(bi, bj) = k \cdot A(\vec{D}) \tag{3.17}$$

where k is the weight to decrease the reliability and it has been set to $k = 0.7$ during the experiments.

3.3.2 Propagating estimates approach

When applying the spatial-temporal approach to real images it is not enough to scan only from left to right and back, because the assumption, that at every point on the edge of an area a correct blur estimate is available, does not always hold. Therefore a method propagating estimates in different directions has been proposed [70]. The idea behind the method is that the reliable estimation will be propagated into a objects more times than outside the object.

The propagating estimates approach starts with creating an edge map of the input image, which indicates where there are accurate blur estimates. The edge map $E(bi, bj)$ is defined as:

$$E(bi, bj) = \begin{cases} 255, & if \ A(bi, bj) \geq A_{th} \\ 0, & else \end{cases} \tag{3.18}$$

where A_{th} is a certain threshold to define strong edges. Eight scan directions are used to create different blur maps according to:

$$\sigma_1(bi, bj) = \begin{cases} \sigma_{(}bi, bj - 1), & if \ E(bi, bj) = 0 \ and \ UV(0, -1) \leq th \\ \sigma_{(}bi, bj), & else \end{cases} \tag{3.19}$$

$$\sigma_2(bi, bj) = \begin{cases} \sigma_{(}bi, bj + 1), & if \ E(bi, bj) = 0 \ and \ UV(0, 1) \leq th \\ \sigma_{(}bi, bj), & else \end{cases} \tag{3.20}$$

$$\sigma_3(bi,bj) = \begin{cases} \sigma_{(}bi+1, bj-1), & if\ E(bi,bj) = 0\ and\ UV(1,-1) \le th \\ \sigma_{(}bi,bj), & else \end{cases}$$

$$(3.21)$$

$$\sigma_4(bi,bj) = \begin{cases} \sigma_{(}bi+1, bj+1), & if\ E(bi,bj) = 0\ and\ UV(1,1) \le th \\ \sigma_{(}bi,bj), & else \end{cases}$$

$$(3.22)$$

$$\sigma_5(bi,bj) = \begin{cases} \sigma_{(}bi-1, bj+1), & if\ E(bi,bj) = 0\ and\ UV(-1,1) \le th \\ \sigma_{(}bi,bj), & else \end{cases}$$

$$(3.23)$$

$$\sigma_6(bi,bj) = \begin{cases} \sigma_{(}bi-1, bj-1), & if\ E(bi,bj) = 0\ and\ UV(-1,-1) \le th \\ \sigma_{(}bi,bj), & else \end{cases}$$

$$(3.24)$$

$$\sigma_7(bi,bj) = \begin{cases} \sigma_{(}bi-1, bj), & if\ E(bi,bj) = 0\ and\ UV(-1,0) \le th \\ \sigma_{(}bi,bj), & else \end{cases}$$

$$(3.25)$$

$$\sigma_8(bi,bj) = \begin{cases} \sigma_{(}bi+1, bj), & if\ E(bi,bj) = 0\ and\ UV(1,0) \le th \\ \sigma_{(}bi,bj), & else \end{cases}$$

$$(3.26)$$

The result is eight different blur maps, where the estimated blur values at edge positions are propagated in different scanning directions. From these eight different blur maps a final blur map needs to be constructed. The blur value of the processed block was assumed valid if all the blur maps had the same value and was discarded otherwise. With eight different scanning directions, this method would only work if all the surrounding edges would have exactly the same blur value, which is not the case in real images. Therefore, a median filter is proposed to find the dominant estimate and remove outliers due to incorrect propagations. Then the final blur estimate is:

$$\sigma_{(}bi,bj) = median(\sigma_1(bi,bj), \sigma_2(bi,bj), ..., \sigma_8(bi,bj)) \qquad (3.27)$$

3.3.3   Segmentation-based blur estimation

The previous two approaches both use the propagation of the reliable estimation. The third approach is based on an object segmentation. The block diagram of the proposed segmentation-based blur estimation approach is shown in Fig. 3.9. Firstly the local blur estimation is performed on the input image and the blur radius on the block level is obtained. Then image features such as blur radius, luminance chrominance, pixel position, are extracted from the image to compose a six-dimensional feature space. A spatially constrained K-means clustering is performed in this six-dimensional feature space. After the convergence of the K-means clustering, the blur radius within every segment is estimated and that results in a segmentation-based blur map.
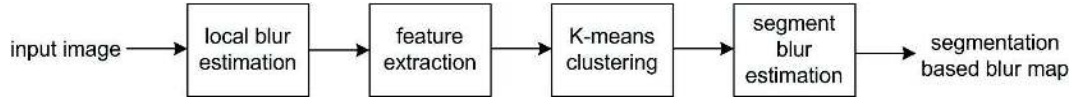
Figure 3.9: The block diagram of the proposed segmentation based blur estimation algorithm.

K-means clustering

We choose the conventional cluster approach K-means because of its good performance and limited complexity. For a brief review of conventional K-means algorithm [84], we suppose that the observation vectors are $\{X_n : n = 1, ..., N\}$. The task for the K-means algorithm is to partition the vectors into K groups with means $\{\mu_k : k = 1, ..., K\}$ such that the total intra-cluster distance is minimized. Every vector is assigned with a cluster label to indicate to which cluster it belongs. We use $l(\cdot)$ to denote the labeling function that maps the feature vector X_n to the cluster number k, denoted as $l(n) = k$. The clustering consists of the following steps:

    1. Specify the number of clusters K according to the requirement and initialize the labels randomly.

    2. Apply iterative steps to update the mean vector μ_k^t in every cluster, and the labels where t is the iteration number.

Calculate the mean vector:

$$\mu_k^t = \sum_{l(n)^{t-1}=k} X_n / N_k^{t-1} \tag{3.28}$$

where X_n is the feature vector and N_k is the number of vectors that belongs to the cluster k.

Update the labels with respect to the minimal distance from the mean vector:

$$l(n)^t = \arg\min_k D(X_n, \mu_k). \tag{3.29}$$

where $D(X_n, \mu_k)$ is the Euclidian distance between X_n and μ_k.

    3. Repeat step 2 until the clustering converges. Convergence, here, means the labels do not change compared to the labels from the previous iteration.

    The disadvantage of K-means is that one has to specify the number of clusters before the clustering. Using a higher number of K will result some over segmentation. However, it is not a problem for the blur estimation. We expect that the over segmented regions will have a similar blur radius and they will merge into a single region in the final result.

| (A) | (B) |
| (C) | (D) |
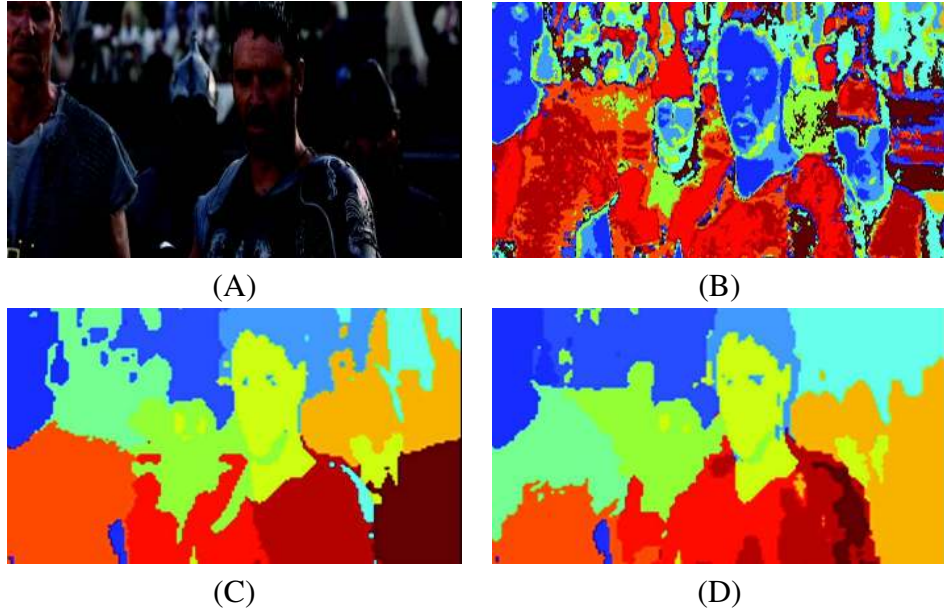
Figure 3.10: Segmentation results using different features: in the segmentation, different color indicates individual segments. (A) the input image (B) using only color information, (C) using color and spatial information, (D) using color, spatial and blur information.

Features

In the K-means clustering, the feature selection is crucial for the successful segmentation. Conventional K-means methods use only color information as features. We believe that the spatial consistency is an essential attribute of objects in sequences. We therefore introduce spatial information into the feature space to achieve a more consistent segmentation. In addition, we also consider the blur radius as an important feature for the object segmentation in images. Eventually, the feature vector that we use includes three components of color information in YUV space, two components of position information and one component of blur radius from the mentioned local blur estimation.

For convenience, we use $X(m), m = 1, ..., 6$ to denote the individual features of a pixel, which horizontal and vertical coordinates in the image are i and j, respectively. Then the three color features are the pixel values $y(i, j)$, $u(i, j)$ and $v(i, j)$ in YUV space, respectively.

$$X(1) = y(i, j). \tag{3.30}$$

$$X(2) = u(i, j). \tag{3.31}$$

$$X(3) = v(i, j). \tag{3.32}$$

To include the spatial information, we add the horizontal and vertical coordinates of the pixel, i and j, in the feature vector:

$$X(4) = i. \tag{3.33}$$

$$X(5) = j. \tag{3.34}$$

Finally, the blur radius $\sigma(i, j)$ at the pixel position is also used as an additional feature.

$$X(6) = \sigma(i, j). \tag{3.35}$$

In order to show the effectiveness of features we selected, we show an example in Fig. 3.10. In the example we perform segmentation using different features. The conventional approach which uses the color information generates a noisy segmentation result (Fig. 3.10 (B)). With the spatial information used as an additional feature, the segmentation result becomes more consistent (Fig. 3.10 (C)). And with the blur radius, the segmentation result becomes more accurate. As the result, the person who is in focus now can be extracted from the background (Fig. 3.10 (D)).

Segmentation blur estimation

After the segmentation, the blur radius within every segment is estimated. In the focused image of an object, a larger blur radius is measured where there is a smooth transition or a flat area on the object surface. The focal blur caused by the optical lens will add a similar amount of blur to every location inside the object boundary. This suggests that the minimal blur radius in the object is caused by the focal blur. Therefore, we propose to assign the blur radius of the segmentation with the minimal blur radius in the segmentation.

3.3.4 Post-processing the final blur map

In order to make the final blur map smoother, we propose to use a bilateral filter for post-processing. The weights for the bilateral filter include the chrominance similarity weight as defined in Equation 3.12 and the spatial weight proposed in the original bilateral filter. The bilateral filter is then defined as

$$\sigma(i, j) = \frac{\sum_{k,l} \sigma(i + k, j + l) \cdot UV(k, l) \cdot S(k, l)}{\sum_{k,l} UV(k, l) \cdot S(k, l)} \tag{3.36}$$

$$S(k, l) = exp[-(k^2 + l^2)/2\sigma_s^2] \tag{3.37}$$

(A)

(B)

(C)
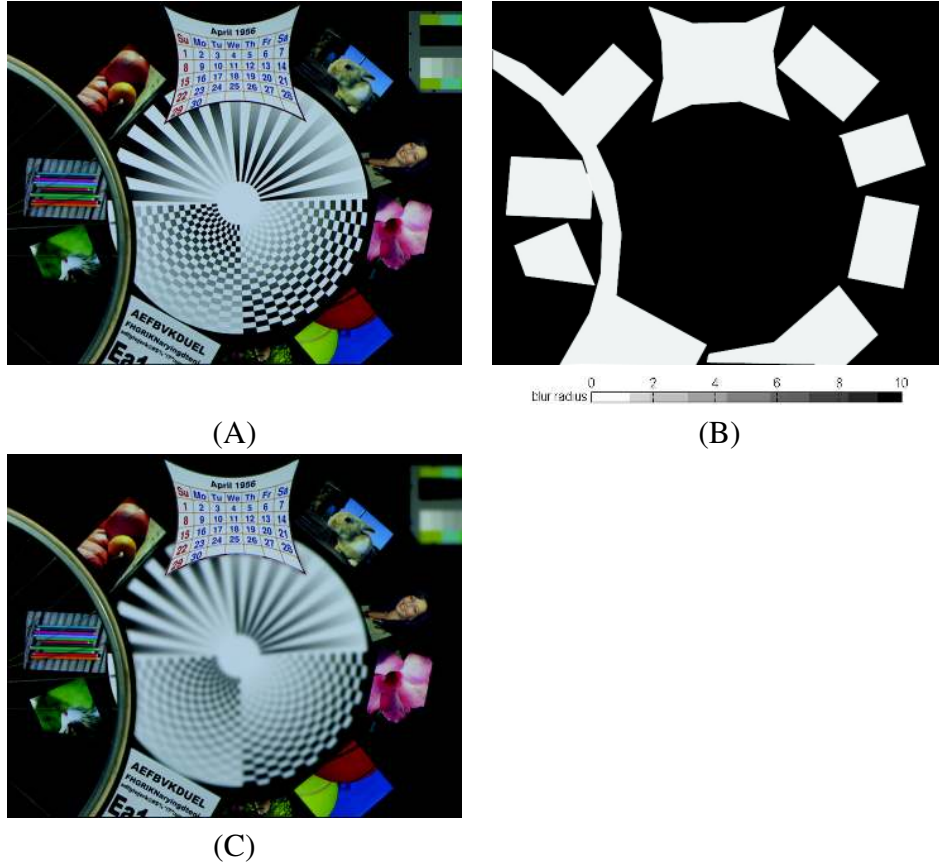
Figure 3.11: The test image used for the evaluation. (A) the all-in-focus image, (B) the ground truth blur map, (C) synthetically blurred image as the simulated input.

where $S(k,l)$ is the spatial weight and σ_s is the standard deviation of the Gaussian function for the spatial weight.

3.3.5   Experimental results

Results on synthetically blurred images

To evaluate the performance of these mentioned estimation approaches, we use an all-in-focus test image shown in Fig. 3.11 (A), which contains various objects. Manually we generate a ground truth blur map shown in Fig. 3.11 (B), which contains the actual synthetical blur value for every pixel position. Based on the ground truth, we synthetically blur the test image to generate a simulated input image shown in Fig. 3.11 (C).

In order to have a quantitative evaluation, we also calculate the mean square

error between the estimated blur maps and the ground truth, as shown in Fig. 3.12. Additionally, we use the average sum of absolute differences between blur values at each pixel position and those at its neighboring pixel positions to indicate the smoothness of the blur estimation. The sum of absolute difference $SAD(i, j)$ at a pixel position (i, j) is defined as:

$$SAD(i, j) = \sum_{k,l} |\sigma(i + k, j + l) - \sigma(i, j)| \tag{3.38}$$

Then the smoothness of the estimate SM is defined as:

$$SM = \frac{\sum\limits_{i,j} SAD(i, j)}{N} \tag{3.39}$$

where N is the total number of the pixels. As we can see, a smaller value of SM means a smoother result.

The results of these proposed methods, including blur maps, MSE and SM scores, are shown in Fig. 3.12. From the results, we can see that the edge-based estimation can already give a very good indication about the actual blur. However, it only appears at the edges. It also generates a noisy result with the smoothness $SM = 20.39$. The spatial-temporal neighborhood approach can fill objects partly with the correct blur values and the overall result is still not very consistent and satisfactory (see the part marked by a rectangle in Fig. 3.12 (B)). The propagation approach improves the performance by filling blur values in some parts inside the objects. However, it generates some wrong propagations at the background area between those focused objects (see the part marked by a rectangle in Fig. 3.12 (C)). The segmentation-based approach has a very consistent blur estimation and generates the smoothest result with the lowest smoothness score $SM = 4.46$. The estimated result matches very well with the actual objects, comparing to the ground truth. This is also reflected in the MSE score. The segmentation-based approach shows the best MSE score at 4.46 and the edge-based approach produces the worst MSE score at 9.24.

Results on natural images

We also test the proposed method on a variety of natural images, including images from compressed MPEG-2 videos and JPEG photo galleries. The images contain different objects, such as flowers, animals, buildings etc. Most of them have complex contents in both focused and unfocused areas. Some of the estimation results are shown in Fig. 3.13 and Fig. 3.14. We do not have the ground truth for these images, therefore, only SM scores are used for indicating the smoothness of the results.
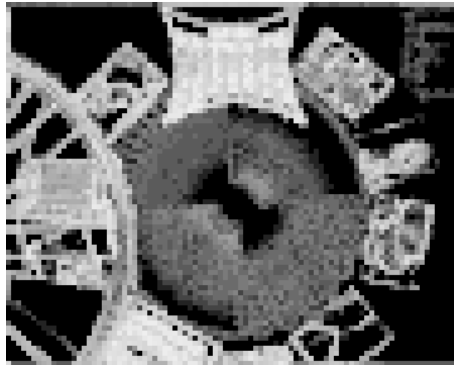
(A) Edge-based approach
MSE=9.24 SM=20.39
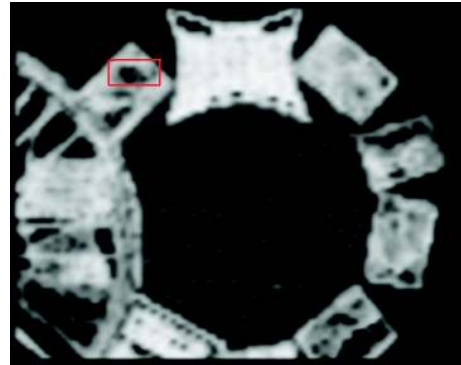
(B) Spatial-temporal neighborhood approach
MSE=5.81 SM=11.52
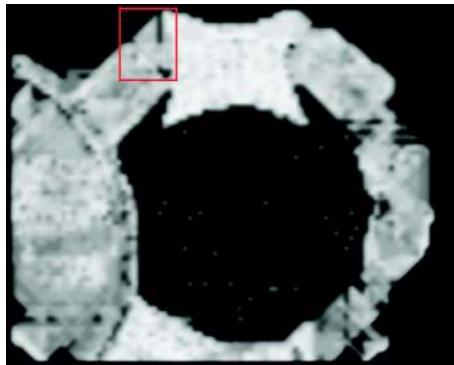
(C) Propagating estimates approach
MSE= 5.01 SM=8.35

(D) Segmentation-based approach
MSE=4.46 SM=6.89

Figure 3.12: The results from different approaches applied on the test image. (A) the edge-based approach, (B) the spatial-temporal neighborhood approach, (C) the propagating estimates approach, (D) the segmentation-based approach.
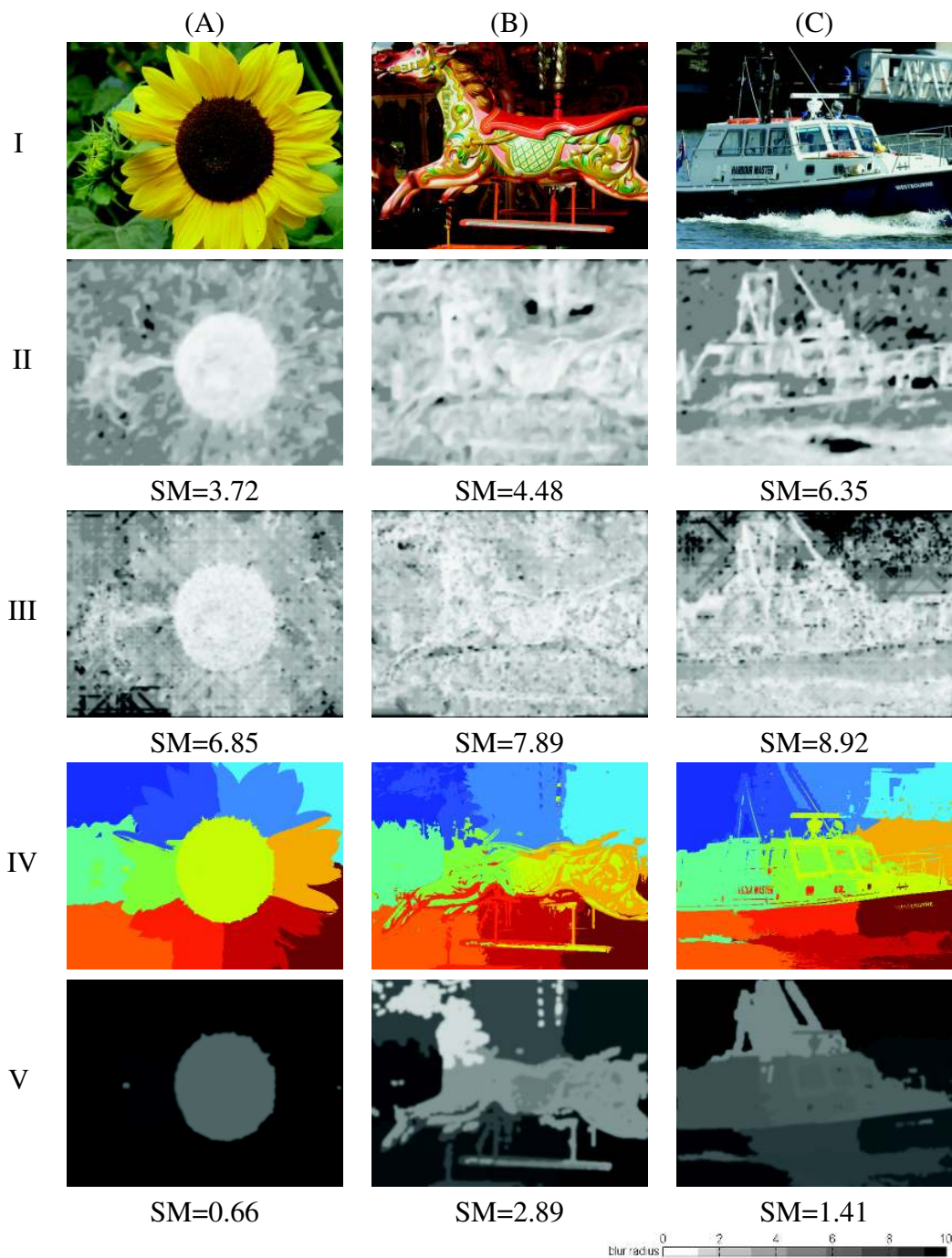
Figure 3.13: Segmentation and blur estimation results from: (I) input image, (II) blur map of the spatial-temporal neighborhood approach, (III) blur map of the propagating estimates approach, (IV) segmentation result, (V) blur map of the segmentation-based approach.
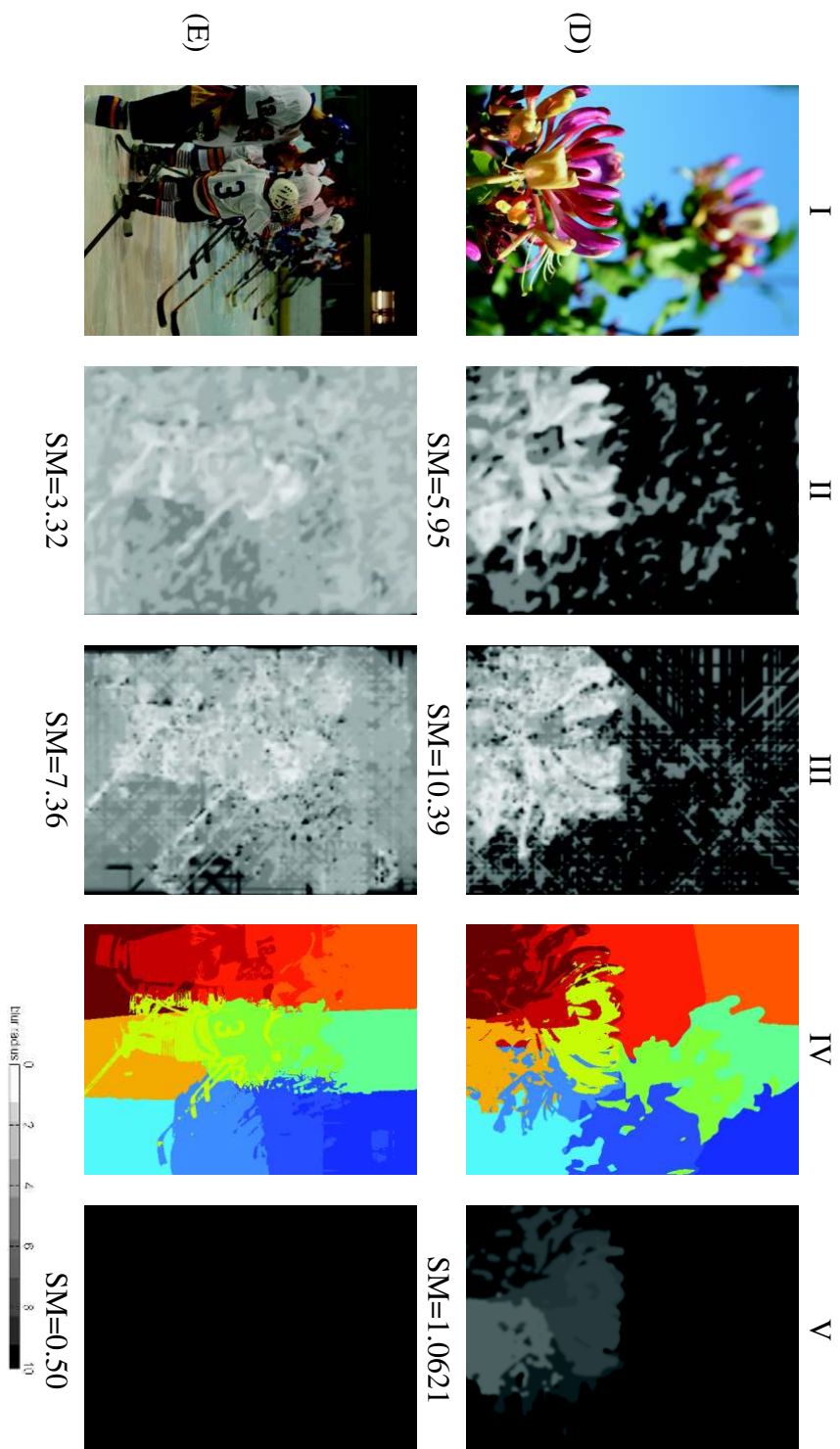
Figure 3.14: Segmentation and blur estimation results from: (I) input image, (II) blur map of the spatial-temporal neighborhood approach, (III) blur map of the propagating estimates approach, (IV) segmentation result, (V) blur map of the segmentation-based approach.

For the smoothness score, the segmentation-based method generates much smoother results than the other two propagation-based methods. For the blur map results, we can see that the two propagation-based methods generate wrong propagations at the background area. Due to more robust estimation within segments rather than the propagation, the segmentation-based method shows better estimation. We see that in most of the results the amount of focal blur has been correctly estimated. In general, the results from the segmentation-based method match well with the perceived region or object segmentation. For instance, the main bodies of the flowers in Fig. 3.13 (A), the horse in Fig. 3.13 (B), and the boat in Fig. 3.13 (C) are all clearly extracted and discriminated from the background. And also in Fig. 3.14 (D) and Fig. 3.14 (E) the same type of objects with different blur has been separated due to the added blur information in the feature space. Although these input sequences or images are compressed and exhibit some coding artifacts, this does not have any impact on the blur estimation because the proposed estimation uses re-blurring which makes the estimation more robust.

In the segmentation results, we can see there are some over segmentation problems. However, these have been partly solved by the blur estimation. By assigning the segmentation blur, the segments with the same blur radius are merged together into one region. For instance, the over-segmented background in Fig. 3.13 (C) and Fig. 3.14 (E). We also see that the over segmentation is not completely solved in Fig. 3.13 (C). It suggests that the segment blur estimation can be further improved. Although the segmentation results are not precise in the pixel level and not the actual object segmentation, they reflect the basic component regions which compose semantic objects or scenes. All reasonable results in the test images indicate that the segmentation and blur estimation are very useful for various applications, such as image layer separation, 2D to 3D video conversion. With specific prior knowledge, the desired semantic objects or scenes can be obtained with these segmentation results easily.

The segmentation based approach has shown the best result among these alternative methods. In the following sections, we will use it in our proposed framework to provide solutions for two applications, focus restoration and blur dependant enhancement.

3.4 Application I : Focus restoration

As objects with varied distances to the camera are differently blurred in the image, it could be interesting to estimate the blur and restore an all-in-focus image. The demand for such a technique is emerging in many applications, such as digital camera and video surveillance. The technique potentially enables the use of algorithms running on relatively cheap DSP chips instead of expensive optical

parts.

In this section, we propose an all-in-focus image restoration based on our segmentation-based blur estimation. Many image restoration techniques [79] use an iterative approach to remove the blur, because they do not need to determine the inverse of a blur operator. However, for real-time applications, iterative approaches are less suitable. Therefore, the LMS filter approach in the proposed framework, regarded as an approximation to the inverse of the blur operation, is more attractive. In order to simultaneously restore the fine structure *and* eliminate the sensor noise, we include local image structure, contrast information and blur radius into the content classification part.

3.4.1 Proposed approach

Fine structures and sensor noise have distinguishable luminance patterns and contrast in natural images. We continue to use adaptive dynamic range coding (ADRC) to classify local image structure. One can see that the fine structures such as edges have regular patterns while the noise shows chaotic patterns. Similar to the contrast classification in Chapter 2, we use 1 bit DR to classify high and low contrast. The threshold for DR is determined by the level of the sensor noise. To combine the blur radius into the classification, we quantize the local blur radius σ obtained from the blur map into a binary RB as:

$$RB = round(\frac{\sigma}{Q}) \tag{3.40}$$

where Q is a predefined quantization step. The concatenation of ADRC code, DR and RB gives the final binary classification code.

The diagram of the proposed focus restoration is shown in Fig. 3.15. The local image content within a 5×5 filter aperture centered at the output pixel is first classified by ADRC, DR and local blur radius at the central pixel position. An LMS filter is used to calculate the output pixel with filter coefficients obtained from the look-up-table (LUT). The filter aperture slides pixel by pixel over the entire image. To avoid an impractical number of classes, we apply ADRC on pixels only in the central 3×3 aperture, that is, we use 8 bits structure information.

The training procedure of the proposed approach is shown in Fig. 3.16. To obtain the training set, we use all-in-focus images as the target images. Furthermore we blur the original image with a Gaussian kernel with a range of blur radii and later add Gaussian noise to simulate the sensor noise at an expected level. These blurred and corrupted versions of the original images are our simulated input images. Before training, the simulated input and the target image pairs are collected pixel by pixel from the training material and are classified using ADRC, DR and
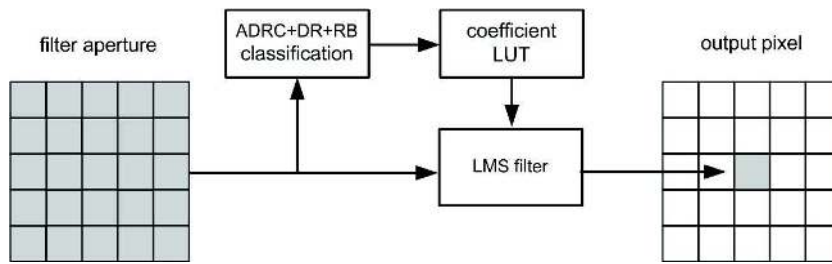
Figure 3.15: The block diagram of the proposed algorithm.

blur radius on the input. The pairs that belong to one specific class are used for the corresponding training, resulting in optimal filter coefficients for this class.
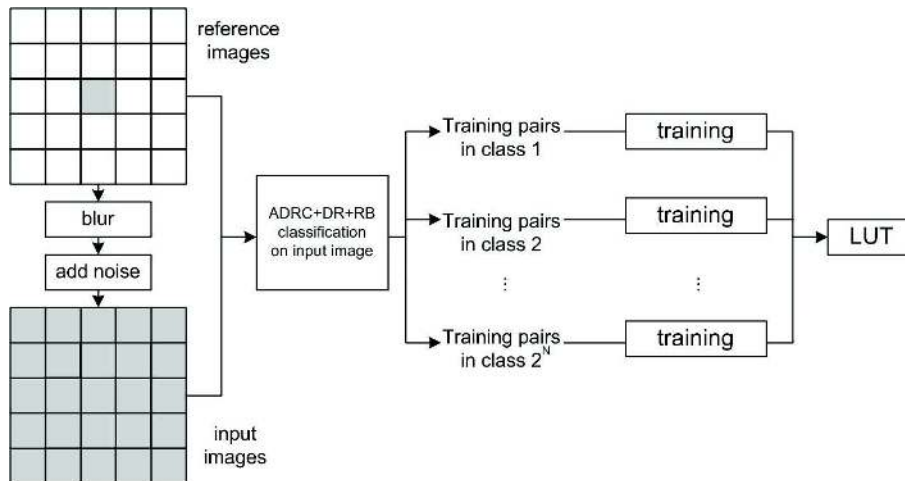
Figure 3.16: The block diagram of the proposed algorithm.

3.4.2 Experimental results

For an objective evaluation, we use some all-in-focus images as the test images, which are not included in the training images. Similar to the training procedure, we blur the test image with a Gaussian kernel with a range of blur radiuses and later add Gaussian noise. Then we apply our proposed method to the synthetically blurred images to get the restored version. The MSE scores between the original images and the restored versions are used for the evaluation. Here, we compare our proposed method with and without estimating the blur radius. Additionally, we investigate different numbers of bits used for the blur classification.
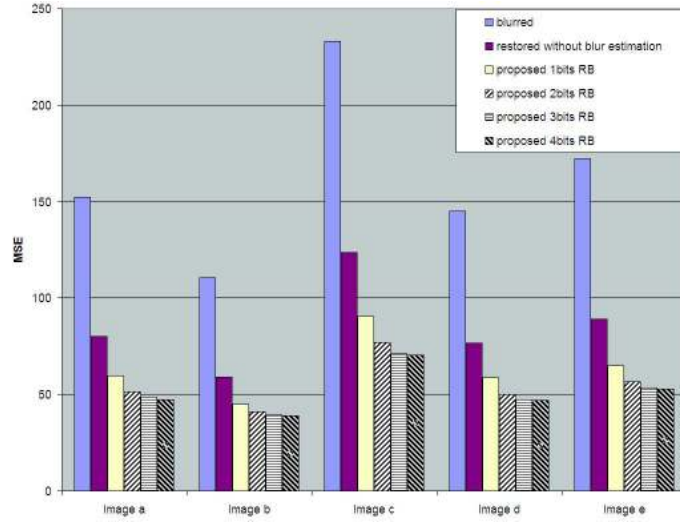
Figure 3.17: Comparison of the MSE scores from the synthetically blurred images and the restored versions: from left to right for each image, the columns show the MSE scores from the synthetically blurred image, the restored version using the proposed method without and with blur estimation.

The MSE comparison is shown in Fig. 3.17. From left to right for each test image, the columns show the MSE scores from the synthetically blurred image, the restored version using the proposed method without blur estimation and the restored version using the proposed method with different numbers of bits for the blur classification. As one can see, the restoration using the proposed method without blur estimation produces lower MSE scores than the blurred inputs. The proposed method further improves the MSE scores with the blur classification. In terms of performance and complexity, we see that using 2 or 3 bits for blur classification could be optimal for the implementation.

To demonstrate the performance of the proposed method on real-world images, we used an image taken by a consumer digital camera as shown in Fig. 3.18, which is not included in the training images. The image shows three objects that are differently blurred. The restored image is shown in Fig. 3.18. Fig. 3.19 shows the blur map estimated by the proposed method. From the estimated blur map, one can see that different blur levels can be clearly discriminated. In the restored image, the focus has been mostly brought back to those differently blurred objects by the proposed restoration. We also see that due to some incorrect blur estimation in the blur map, there are some blurred areas between focused and un-focused objects, which have not been restored. It suggests that in the future research, the accuracy of the blur map should be further improved. Additionally, we compare the proposed method with and without blur estimation. Fig. 3.20 shows
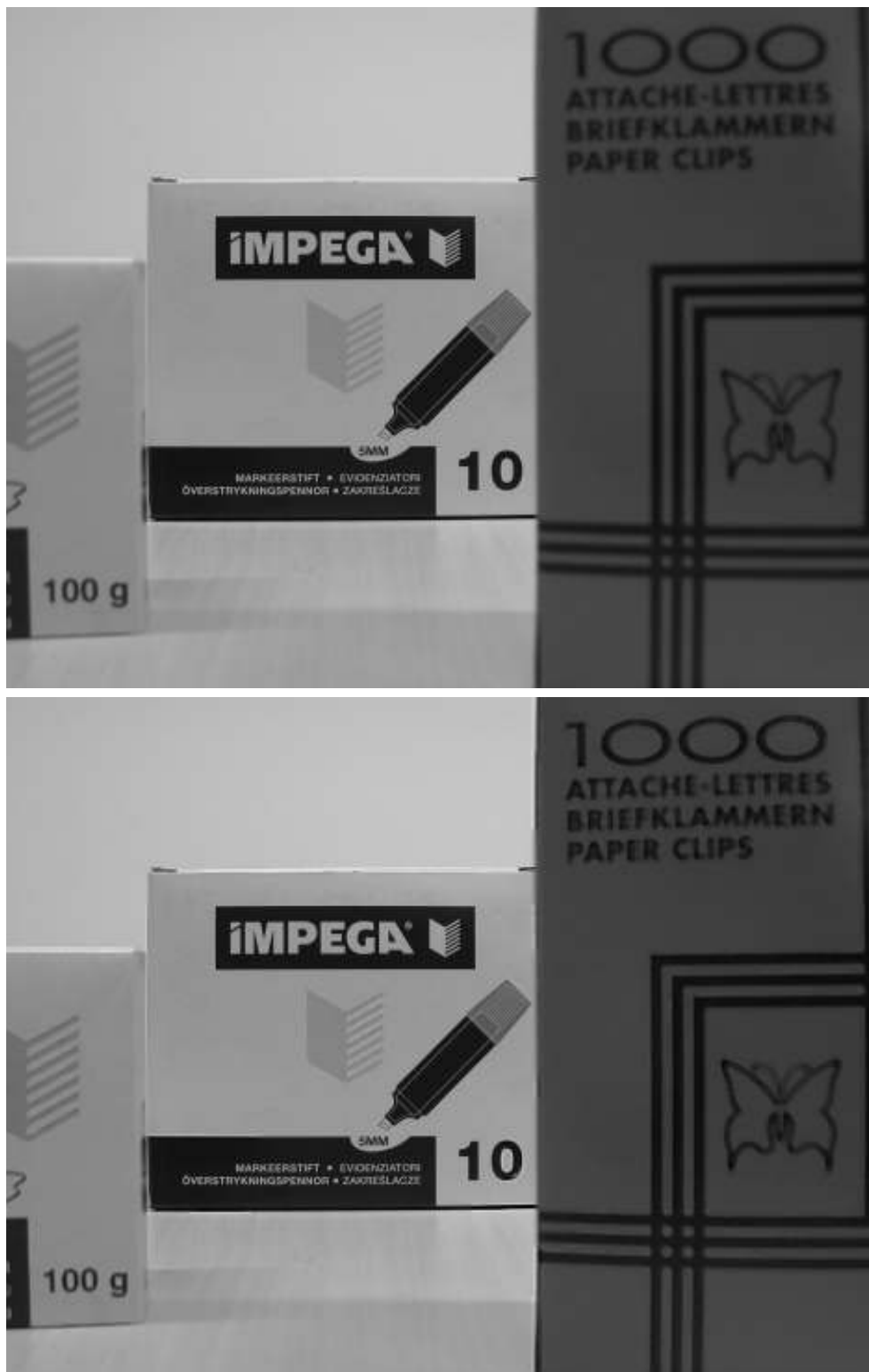
Figure 3.18: Top: the test image with differently blurred objects taken by a digital camera. Bottom: the restored all-in-focus image.
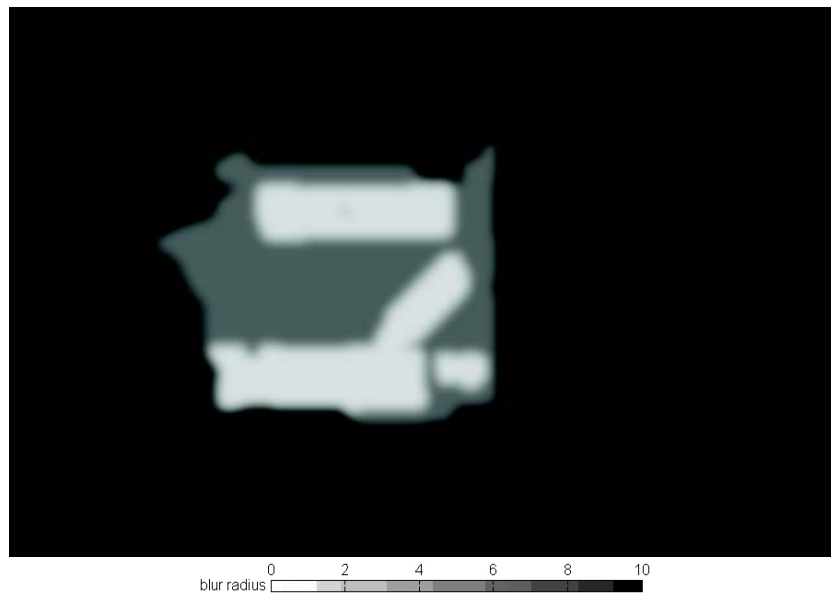
Figure 3.19: The blur map used for focus restoration.

(A)   (B)   (C)
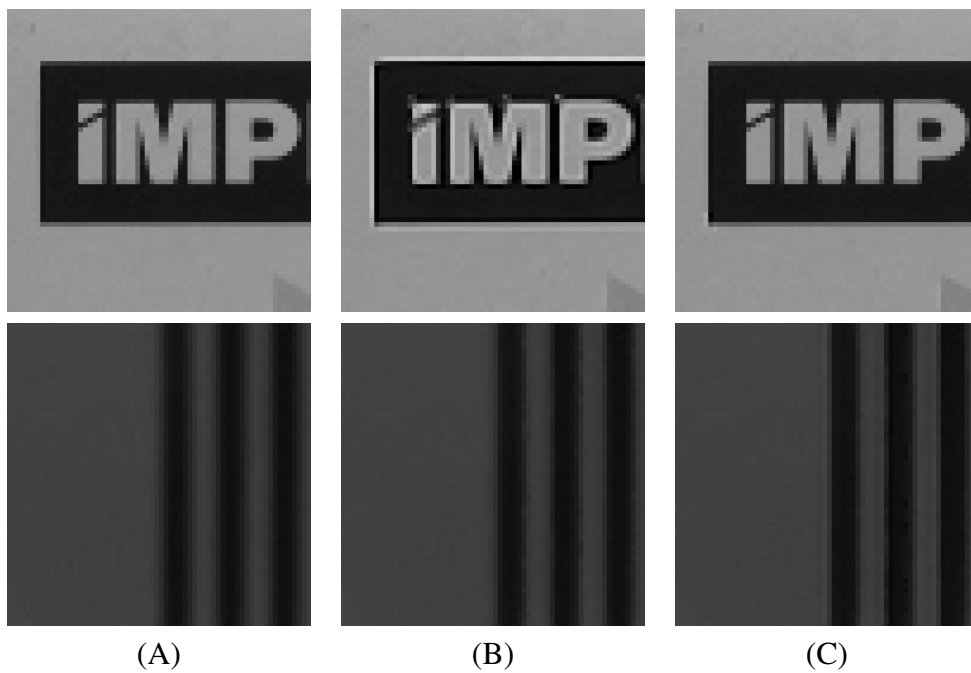
Figure 3.20: Image fragments from both focused and unfocused parts in the restored image:
(A) the test image, (B) restored by the trained filter without blur estimation, (C) restored by the
proposed approach.

image fragments from both focused and unfocused parts in the test image. One can see that the focus part in the test image has been detected with a small blur radius so that the proposed method does not apply much restoration. The result from the proposed method without blur estimation shows incorrect restoration and causes some overshoots at edges. For the unfocused part, the proposed approach with blur estimation also demonstrates better restoration than that without blur estimation.

3.5 Application II : Blur dependent coding artifacts reduction

For digitally coded images or videos, coding artifacts are more visible in smooth areas [59], for example, out of focus areas. Therefore, strong coding artifact reduction is encouraged in these areas. For other areas like smooth transitions in the focused part, modest coding artifact reduction is more likely appreciated to avoid over-smoothing effects. We expect that using the additional blur information, better artifact reduction and sharpness enhancement can be achieved. In Chapter 2, we have presented an approach for simultaneous artifact reduction and sharpness enhancement. However, the classification only considers the local structure and contrast. Therefore, to further improve the enhancement, we add the blur information to the content classification.

3.5.1 Proposed approach

To combine the blur radius into the classification, we quantize the local blur radius σ obtained from the blur map into a binary RB as the previous section. The concatenation of ADRC code, DR and RB gives the final classification code.

The training procedure is similar to the enhancement mentioned in Chapter 2. To obtain the training set, we use high quality images as the reference output images. Then we blur the original images with a small blur radius to introduce the sharpness enhancement and then compress them to generate coding artifacts. These blurred and corrupted versions of the original images are our simulated input images. The proposed segmentation based blur estimation is applied on the input images to obtain the blur map.

3.5.2 Experimental results

To show the performance of the algorithm, we use the test images in Fig. 3.13 - Fig. 3.14. These images are blurred and compressed using JPEG compression

Table 3.1: MSE scores of different enhancement.

Image	Mean Square Error					
	input	enhancement in chapter 2	Proposed 1bit RB	Proposed 2bit RB	Proposed 3bit RB	Proposed 4bit RB
(A)	35.0	30.7	29.3	29.1	28.9	28.8
(B)	81.7	65.7	63.9	63.1	62.6	62.4
(C)	49.3	38.7	37.1	36.7	36.4	36.3
(D)	19.1	15.7	14.3	14.1	13.8	13.8
(E)	21.3	16.9	16.0	15.8	15.6	15.5
Average	41.2	33.5	32.1	31.8	31.5	31.4

as in the training procedure. Then we apply the proposed enhancement with and without blur estimation on the simulated input images. Then the MSE between the original and the enhancement version can be used for the evaluation. Similar to the evaluation in the application of focus restoration, we also investigate different numbers of bits used for the blur information. In addition to the MSE score, we also used the BIM metric proposed by Wu [101] for the evaluation. The BIM metric measures the blockiness of compressed images or sequences. Value $BIM = 1$ refers to no blockiness at all and the larger the BIM value is, the more blockiness in the content.

The comparison of the MSE scores from the proposed enhancement with and without blur estimation is shown in the Table 3.1. We can see that with blur estimation the proposed approach can achieve a better MSE score. Because the artifacts in the large blurred area cause a numerically small error, the improvement on the MSE score is not that significant. For the BIM scores in Fig. 3.21, we see that including the blur information clearly improves the artifact reduction. Given both the MSE and BIM scores, we can see that the optimal number of bits for BR is 3.

To have a qualitative comparison, the result of the test image in Fig. 3.13(A) is shown in Fig. 3.22. We can see that the image has been well enhanced by the proposed method. The coding artifacts have been greatly smoothed and the flower in focus is nicely enhanced. In order to show the effectiveness of the blur estimation, image fragments from both focused and unfocused parts from the results of the proposed enhancement with and without blur estimation are shown in Fig. 3.23. The proposed enhancement has shown a better ability to remove the blocking artifacts in the unfocused part and has a better sharpness enhancement in the focused part.
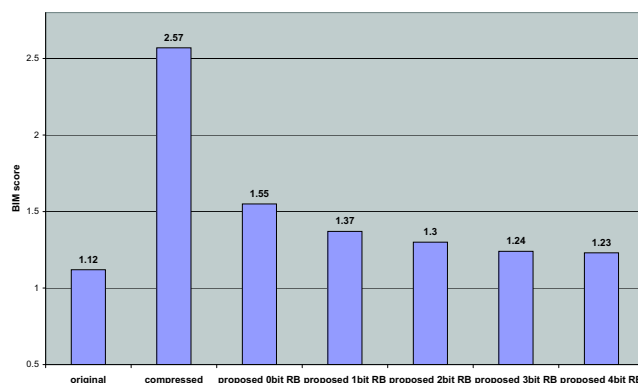
Figure 3.21: The BIM scores of different enhancement methods.

3.6 Conclusion

In this chapter, we have first presented a novel robust, low-cost blur estimation algorithm based on image edges. The maximum of the difference ratio between an original image and its two re-blurred versions has been proposed to identify the edges and estimate the local blur radius in the original image. The proposed method has been shown to have robust estimation, especially for the interference from neighboring edges.

To generate more consistent blur estimation and discriminate focal blur from other types of non-degradation blur, we have proposed a new segmentation-based blur estimation method. The spatial constraints and blur information have been introduced to the feature space to achieve a more robust and accurate object segmentation and blur estimation, which has been tested on a variety of images. Estimating the segment blur also alleviates the over-segmentation problem. Further we have applied the proposed segmentation based blur estimation to the proposed framework, which results solutions to two applications, focus restoration and blur dependent coding artifact reduction. The experiment results have shown that the proposed blur estimation is very useful and brings significant improvement to these applications.

Figure 3.22: Blur dependant enhancement on the image at Fig. 3.13(A). Top: the input compressed image. Bottom: the enhancement image with the proposed blur depend enhancement.
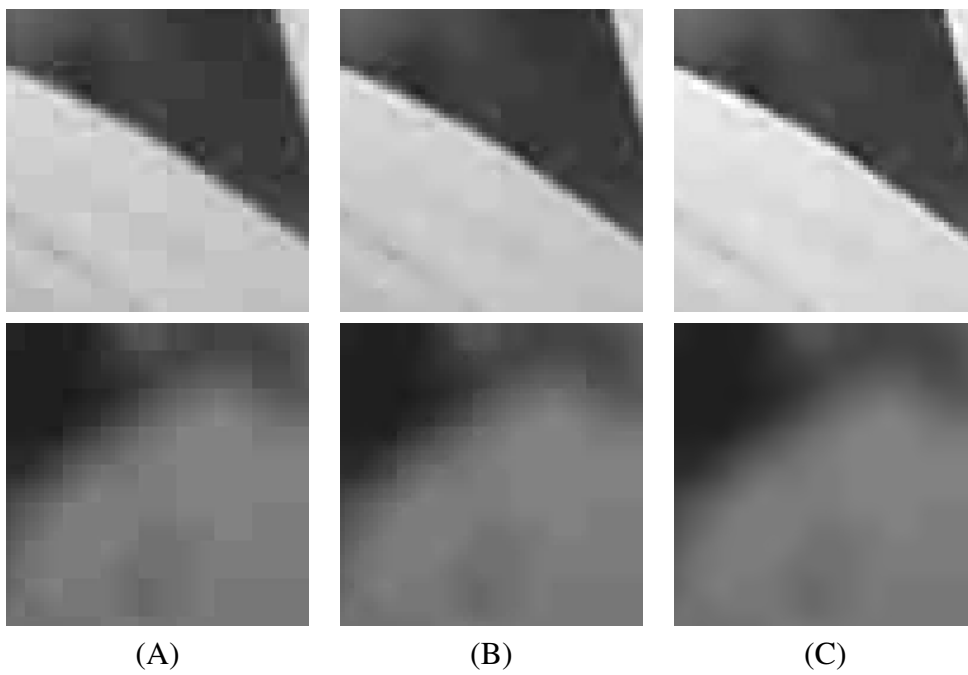
Figure 3.23: Image fragments from both focused and unfocused part from the enhancement result of on the image at Fig. 3.13: (A) the simulated input image, (B) the result of the enhancement without blur estimation, (C) the result of the proposed enhancement.

Chapter 4

Class-count Reduction

In the previous chapters, we have introduced different classifiers into the content classification part of the proposed content adaptive enhancement framework, which has demonstrated superior performance for many video enhancement applications. The content classification starts with the local structure classification for resolution enhancement, such as edge direction and adaptive dynamic range coding for luminance patterns. However, only local structure classification is not enough for classifying image content in many other applications. We have seen in the application to coding artifact reduction, the combination of local structure and *local contrast* classification is necessary for good performance. Adding more classifications such as local variance further improves the performance. In the application to focus restoration, including robust local blur classification can adaptively enhance differently blurred parts in an image.

We expect that more content classification will be explored in the future. As a convenient and scalable algorithm design, incorporating more features in the classification widens the scope of the proposed framework for more possible applications and improves the overall performance, but also leads to an explosion of the class-count (the number of the filters), many of which may be redundant.

For hardware implementation, it would be desirable to have some class-count reduction techniques in which users could specify the number of classes to allow a graceful degradation of the performance. Early approaches like Atkins' method of resolution synthesis [105] provide a way to obtain a flexible number of classes from the image structure. However, it is originally designed only for feature classification in resolution enhancement and is computationally expensive. In this chapter, we propose three class-count reduction techniques, *class-occurrence frequency*, *coefficient similarity* and *error advantage*, which are based on our proposed framework and are generally applicable for different content classifications. The results show that with these techniques the number of classes can be greatly reduced without serious performance loss.

The rest of this chapter is organized as follows. We begin with a brief review about two classification based filtering approaches in Section 1. In Section 2, three different class-count reduction techniques are presented. The evaluation of the techniques in the applications of resolution up-scaling and coding artifact reduction are shown in Section 3. Finally, in Section 4, we draw our conclusions.

4.1   Introduction

The classification based filtering was first proposed by Kondo's method [97] for image interpolation, where only the image structure such as edge direction or luminance pattern is used for the classification. In this thesis it has been extended to a more general framework for more applications, such as coding artifacts and blur dependant enhancement. The usual scheme in such methods is that, an input pixel vector X consisting of N pixel values from the local image content within a filter aperture is first classified by image features, such as local structure and contrast, and a LMS optimal linear filter for that class is used to estimate the output pixel $\hat{y}$ with filter coefficients from a look-up-table (LUT). The LMS filter coefficients are determined in an off-line training using simulated input and reference output images. The filtering process can be defined as:

$$\hat{y} = \sum_{j=1}^{M} L(j, X) W_j^T X. \tag{4.1}$$

$$L(j, X) = \begin{cases} 1, & if \quad X \quad belongs \quad to \quad class \quad j \\ 0, & otherwise \end{cases} \tag{4.2}$$

where W_j is the filter coefficient vector for class j and $L(j, X)$ is a function indicating whether vector X belongs to class j.

Such content adaptive filtering techniques can be categorized as *hard classification* based methods since the classification result is always 1 or 0. Instead of hard classification, some methods like Atkins' method [105] use *soft classification*, that is, the probability of a given input vector x belonging to a certain class as the classification output. In Atkins' method, the input vectors are modeled as a Gaussian mixture with M classes, where every class corresponds to a multivariate Gaussian model with a parameter θ. The expectation maximization (EM) algorithm [83] is applied to compute the maximum likelihood (ML) estimates of the Gaussian model parameters θ. The probability of a given input vector X belonging to class j, $p(j|X)$, is computed with the parameters θ. In every class, a probability-weighted LMS algorithm is used to estimate the corresponding high resolution pixels. Final high resolution pixel estimates are computed as a weighted

average of the estimates for all classes, where the weights are the probabilities of the input vector belonging to every class.

The derivation of Atkins' method starts with the following assumptions:

Assumption 1:

The input pixel vectors X is modeled as a multivariate Gaussian mixture. That is, the probability density function $p_X(X)$ is:

$$p_X(X) = \sum_{j=1}^{M} \pi_j p_{X|j}(X|j), \quad p_{X|j}(X|j) \sim \mathcal{N}(\mu_j, \sigma_j) \tag{4.3}$$

where j is the mixture class index; π_j is the probability of cluster j; $p_{X|j}(X|j)$ is a multivariate Gaussian density; μ_j is a N dimensional mean vector and σ_j is a $N \times N$ covariance matrix. The Gaussian density $p_{X|j}(X|j)$ is:

$$p_{X|j}(X|j) = (2\pi)^{-M/2} |\sigma_j|^{-1/2} exp(-1/2(X - \mu_j)^T \sigma_j (X - \mu_j)) \tag{4.4}$$

It is found [105] that choosing the covariance for all the classes to be $\sigma^2 I$, where I is the identity matrix, improves the performance. Therefore, Equation 4.4 becomes

$$p_{X|j}(X|j) = (2\pi)^{-M/2} |\sigma^2 I|^{-1/2} exp(\frac{-1}{2\sigma^2} ||X - \mu_j||^2) \tag{4.5}$$

Assumption 2:

Given the input low resolution pixel neighborhood and the context class, the high resolution pixel is multivariate Gaussian. Given the input vector X, the class distribution is independent of the high resolution and low resolution pixels.

By these assumptions, the minimal MSE estimate [82] will be:

$$\hat{y} = \sum_{j=1}^{M} (A_j X + \beta_j) p_{j|X}(j|X). \tag{4.6}$$

According to Bayes' rule, the posterior probability $p_{j|X}(j|X)$ will be:

$$p_{j|X}(j|X) = \frac{\pi_j p_{X|j}(X|j)}{\sum_{j=1}^{M} \pi_j p_{X|j}(X|j)} \tag{4.7}$$

By inserting Equation 4.7 into Equation 4.6, the equation of optimal image inter-

polation is obtained:

$$\hat{y} = \sum_{j=1}^{M} (A_j X + \beta_j) \frac{\pi_j p_{X|j}(X|j)}{\displaystyle\sum_{j=1}^{M} \pi_j p_{X|j}(X|j)}$$

$$= \sum_{j=1}^{M} (A_j X + \beta_j) \frac{\pi_j exp(\frac{-1}{2\sigma^2}||X - \mu_j||^2)}{\displaystyle\sum_{j=1}^{M} \pi_j exp(\frac{-1}{2\sigma^2}||X - \mu_j||^2)} \tag{4.8}$$

The output pixel is computed as a combination of the output of all M filters. In practice, it is more computationally efficient to combine the filter coefficients first. Let

$$A_j\prime = \sum_{j=1}^{M} A_j \frac{\pi_j exp(\frac{-1}{2\sigma^2}||X - \mu_j||^2)}{\displaystyle\sum_{j=1}^{M} \pi_j exp(\frac{-1}{2\sigma^2}||X - \mu_j||^2)} \tag{4.9}$$

$$\beta_j\prime = \sum_{j=1}^{M} \beta_j \frac{\pi_j exp(\frac{-1}{2\sigma^2}||X - \mu_j||^2)}{\displaystyle\sum_{j=1}^{M} \pi_j exp(\frac{-1}{2\sigma^2}||X - \mu_j||^2)} \tag{4.10}$$

Then Equation 4.8 becomes:

$$\hat{y} = A_j\prime X + \beta_j\prime. \tag{4.11}$$

In Atkins' method, the number of classes can be selected independently from the filter aperture. In the hard classification based methods that we used throughout this thesis, however, they are often directly related. For instance, the number of ADRC classes and filter coefficients increases exponentially with the number of pixels in the aperture. Every bit added for additionally coding the contrast, blur, or grid position, further doubles the class count. A flexible number of classes, as in Atkins' method, would be desirable. Therefore, the goal is to design an algorithm that uses hard classification and allows a flexible number of classes.

4.2   Class-count reduction techniques

As one can see in the previous section, the disadvantage of using hard classification in the proposed framework is that it can introduce a large number of classes.

Table 4.1: The contributions from the occurrence frequency sum of the first n most frequently occurring classes to all the content classes in an image database.

n	64	128	256	512	1024	2048
Contribution	78.52%	87.96%	93.06%	96.32%	98.46%	99.59%

With such a large number, the method may not be efficient, i.e., there may be some redundancy in the classes. In this section, we explore three clustering techniques all capable to reduce the total number of classes, namely, class-occurrence frequency, coefficient similarity and error advantage. These techniques all use a similar scheme as follows. First one or more content classes will be clustered in a class-cluster. In every class-cluster an optimal linear filter is used. Every content class is assigned with a class-cluster label to indicate to which class-cluster it belongs. We use $f(\cdot)$ to denote the labeling function that maps the content class k to the class-cluster number j.

The filtering process is shown in Fig. 4.1. First the input vector from the local image content will be classified into content classes using different features, then the label look-up-table (LUT) is used to find out which class-cluster the content class belongs to. Then the corresponding filter is chosen for computing the output.
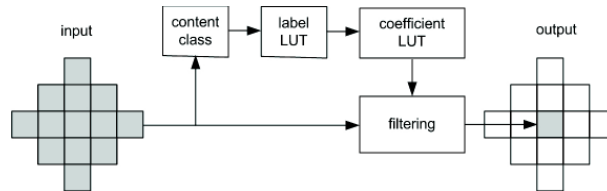
Figure 4.1: The block diagram of the class-reduced algorithm: the input vector is first pre-classified using the content classification, then the content class is used to get the cluster number from the label LUT. Finally, the filter coefficients for the cluster are used for the processing.

4.2.1 Class-occurrence frequency (CF)

One way to reduce the number of classes is to merge the classes which are less important for the perceived image quality. The importance of a class is likely reflected to how often it occurs in an image. Here, we use how many times a class has occurred as the occurrence frequency of that class. Then we could count the occurrence frequency of every content class and hope that it can tell how important the content class is for the perceived image quality. For instance, in the

application of coding artifact reduction application mentioned in Chapter 2, there are 8192 classes with the classification of local structure and contrast. However, the contribution from every class is not equal. We would expect that some classes will occur in a natural image much more frequently than others. We then count the occurrence frequency of every class and list the contributions from the occurrence frequency sum of the first n (n=64, 128, 256, 512, 1024, 2048) most frequently occurring classes to all the content classes in an image database as shown in Table 4.1. We see that the first 2048 most frequently occurring classes already count for more than 99 percent of all the content. There are so many classes that rarely occur in the content.

We would expect that if we merge the rarely occurring classes, the overall performance will suffer little. Suppose X_k denotes all the input vectors in content class k. We sort the input vectors $X_1, X_2, ..., X_k$ to $X_{[1]}, X_{[2]}, ..., X_{[k]}$ from high to low by the occurrence frequency. The M least frequent content classes will be merged into a cluster. For the most popular classes, each will remain as a separate cluster. The labels will be:

$$f(i) = \begin{cases} [i], & if \quad [i] < M \\ M, & otherwise \end{cases} \tag{4.12}$$

In this reduction technique, only one cluster includes more than one content classes. Therefore, in the hardware implementation, a number of comparators can be used instead of the more expensive label LUT. Fig. 4.2 shows a block diagram of using such comparators. $M - 1$ comparators contain the class codes $C_{[1]}, C_{[2]}, ..., C_{[M-1]}$ which are sorted by its occurrence frequency. Once the input pixels are classified by different content classifications, the class code will be compared with the $M - 1$ most frequently occurring class codes. The comparison results are combined to a binary code to address the coefficient LUT.
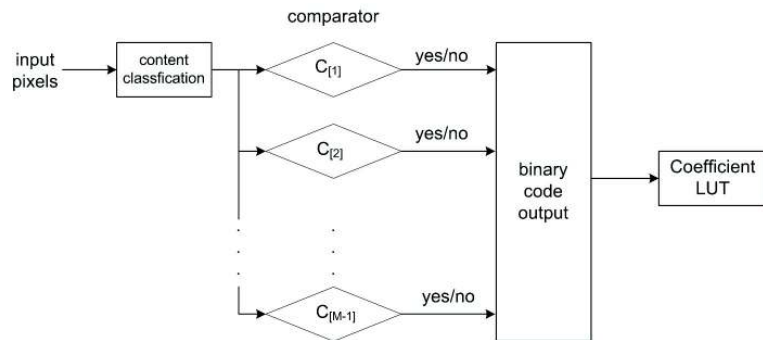
Figure 4.2: The block diagram of using comparators for class-count reduction: $M - 1$ parallel comparators which contain the class codes sorted by their occurrence frequencies are used.

4.2.2 Coefficient similarity (CS)

Another possible way to reduce the class-count is to examine the similarity between the filter coefficients obtained from the training in every content class. The filter coefficients directly show the filtering behavior and classes with similar coefficients can be merged likely with limited effect on the performance. The similarity, here, is indicated by the Euclidian distance between coefficient vectors. We propose to use the K-means algorithm [84] to cluster the classes.

The clustering consists of the following steps:

1. Specify the number of clusters M according to the requirement and initialize the labels randomly.

2. Apply iterative steps to update the mean vector μ_j^i in every cluster and all the labels, where i is the iteration number and j is the cluster number. Calculate the mean vector:

$$\mu_j^i = \sum_{f(k)^{i-1}=j} W_k/N_j^{i-1} \tag{4.13}$$

where W_k is the coefficient vector from the content class k and N_j is the number of content classes that belongs to the cluster j.
Update the labels with respect to the minimal distance from the mean vector:

$$f(k)^i = \arg\min_j D(W_k, \mu_j). \tag{4.14}$$

where $D(W_k, \mu_j)$ is the Euclidian distance between W_k and μ_j.

3. Repeat step 2 until the clustering converges. Convergence, here, means the labels do not change compared to the labels from the previous iteration.

4.2.3 Error advantage (EA)

As we can see, the content adaptive methods try to minimize the mean square error (MSE) between the output image and the target images, in every class. These two mentioned class-count reduction techniques so far are based on heuristics, i.e., the classification they employ does not guarantee the minimization of the total intra-cluster MSE. To achieve just that, we propose a third technique that clusters the content classes with respect to the error advantage of cluster LMS filters. Thus, the minimal total MSE can be achieved given a fixed number of clusters and a particularly bigger set of classes.

Atkins' method classifies the local content of the low resolution image by assuming the input vector is a multivariate Gaussian. However, that assumption is not very strong. We expect that if the classification is performed without the reference data, it mat not be optimal for fitting the image data into linear models.

In contrast, we propose to include the reference images to improve the clustering, which should lead to better processing results.

Therefore, a clustering scheme illustrated in Fig. 4.3 to minimize the total intra-cluster MSE is proposed. The clustering scheme consists of the following steps:
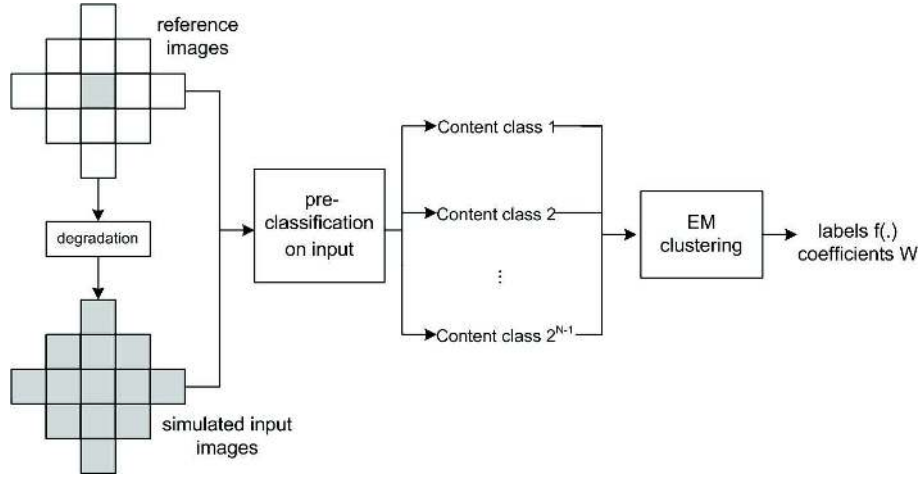
Figure 4.3: The clustering procedure of the proposed error advantage approach. The input and output pairs are collected from the training material and are pre-classified into a number of content classes. The EM algorithm is used to cluster the content classes to minimize the intra-cluster LMS error.

1. Build the training set. The vector pairs that consist of input and reference pixels are collected from the simulated input and reference images, respectively. S_k denotes the collection of all the vector pairs whose input vectors belong to content class k. Specify the number of clusters M according to the requirement and initialize the labels randomly.

2. Apply the Expectation Maximization iterations [83] to the content classes to update LMS filter coefficients CW_j^i of all the clusters and cluster labels $f(\cdot)^i$, where i is the number of iterations. The total intra-cluster MSE will decrease after every iteration until the clustering converges.

M-step: Obtain the cluster LMS filter coefficients CW_j^i by the LMS algorithm using the labels $f(\cdot)^{i-1}$ as in Equation 1.3.

$$CW_j^i = (\sum_{f(k)^{i-1}=j} S_{k,x} S_{k,x})^{-1} \sum_{f(k)^{i-1}=j} S_{k,x} S_{k,y} \qquad (4.15)$$

where $S_{k,x}$ and $S_{k,y}$ denote all input vectors and reference vectors from the content class k, respectively.

E-step: Evaluate the coefficient vector CW_j^i on every content class and update the labels of content classes with respect to the minimal error.

$$f(k)^i = \arg\min_j E[(S_{k,y} - CW_j^{iT}S_{k,x})^2] \tag{4.16}$$

3. Repeat step 2 until the labels no longer change compared to the labels from the previous iteration.

Comparing to the coefficient similarity approach, the iteration here involves much more calculations to evaluate all the cluster coefficients on the whole training set. Therefore, higher computation load and more training time is expected for the error advantage approach.

4.3 Algorithm complexity analysis

In order to have some indication of the implementation cost of these classification based algorithms (original Kondo's method, Atkins' method and the proposed method), we perform a modest complexity analysis in this section.

As we can see from previous sections, all these three algorithms share a similar block diagram. First an input pixel vector x consisting of N pixel values from the local image content within a filter aperture is classified. Next, filter coefficients from the off-line training will be fetched to compute the output pixel. The calculation of the current output pixel does not depend on the previous output pixel, or every output pixel can be calculated without other output pixels. This means they are all equally suitable for pipeline processing and these three algorithms have the same degree of data parallelism. For the input pixel buffer size, the pixel vector is the only data needed from the input image. Therefore, all these three algorithms have the same pixel buffer size, which depends on the filter aperture size.

A further common part of all these three algorithms is that they have only one linear filter for every pixel in the end. Therefore, we could compare the cost to obtain the filter coefficients for each case. For Kondo's method, the ADRC class code is computed first, which consists of N comparisons and $2N$ bit operations. Then N coefficients for a given class are fetched from the coefficient look-up-table. For the proposed method, the operations are similar, except that an extra fetch from the label look-up-table is needed. For Atkins' method, all cluster parameter (μ) and filter coefficients (A, β) need to be fetched for classification. The calculation of the posterior probability (Equation 4.7) includes $N \times M$ additions, $3N \times M$ multiplications and M exponential operations. The combination of the filter coefficients includes $N \times M$ additions and $(N + 1) \times M$ multiplications.

Table 4.2 lists the complexity comparison of obtaining coefficients for the final linear filter per output pixel for all the algorithms. In addition, it also shows

Table 4.2: Algorithm complexity analysis.

Algorithm	Kondo's	Atkins'	Proposed
Fetch operations	N	$(2N + 2) \times M$	$N + 1$
Comparison operations	N	0	N
Bit operations	$2N$	0	$2N$
Multiplication operations	0	$4N \times M$	0
Addition operations	0	$2N \times M$	0
Exponential operations	0	M	0
Coefficient numbers	$N \times 2^N$	$(2N + 2) \times M$	$N \times M + 2^N$

the total number of coefficients for every algorithm. From the table, we can see that the proposed method has a similar complexity with Kondo's method, while Atkins' method has a much higher complexity. Atkins' method is computationally far more expensive because the probabilities need to be calculated and the final filter coefficients are mixed on-the-fly, where the filter coefficients for the proposed and Kondo's methods are directly available in the pre-obtained LUT. In terms of the number of coefficients, the proposed method is the lowest, given the fact that the setting $N > 9, M < 100$ is usually used. This also suggests that the filter coefficients of the proposed method can more likely be stored in the cache of a processor in order to speed up the process.

4.4   Experimental results

In this section, we will evaluate these three class reduction techniques, class-occurrence frequency (referred to as CF), coefficient similarity (referred to as CS) and error advantage (referred to as EA), in the applications of coding artifacts reduction and image interpolation.

4.4.1   Application to coding artifact reduction

We start with the application of simultaneous coding artifact reduction and sharpness enhancement proposed in Chapter 2. In that application, we conclude that ADRC classification alone is not enough to distinguish between the coding artifacts and real image structures. Therefore, one extra classification describing the contrast information in the filter aperture is added. Here we use the same filter setting as in Chapter 2. The filter aperture is a diamond shape consisting of 13 pixels. An extra classification bit is used for the local contrast classification. The total number of classes is 8192.

In the experiment, we use a training set including about 2000 high resolution (1920 by 1080) frames. For the evaluation, we use some test sequences shown in Fig. 4.4, which are not included in the training set. The test sequences are first blurred, then compressed, as in the training, to generate the simulated input sequences. Then the proposed method in Chapter 2 with all these three class-count reduction techniques is evaluated using these sequences. For a fair comparison, we use the same number of clusters for these three class-count reduction techniques. An attractive number for hardware implementation, $M = 32$, is chosen for the experiment. For reference, we also include a fixed LMS filter which uses no classification.

(A) Bicycle (B) Lena

(C) Birds (D) Boat (E) Motor
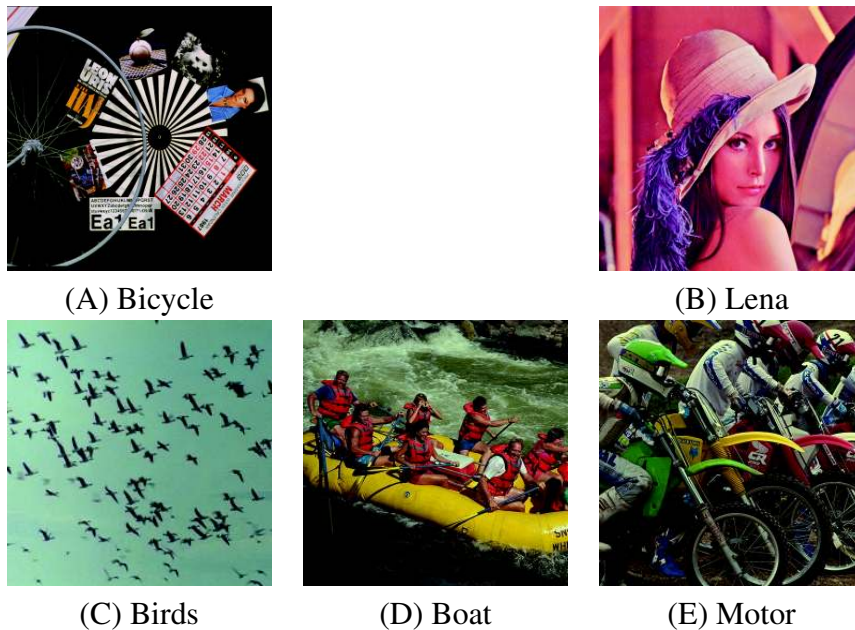
Figure 4.4: The testing material used for the evaluation in coding artifact reduction.

Table 4.3 shows the MSE comparison of the evaluated methods. In terms of the MSE score, one can see that these three reduction techniques can reduce the number of classes by a factor of 256 with a modest increase of the MSE, compared to the MSE increase by using the fixed LMS filter. Among these three techniques, EA achieves the lowest MSE score, which is expected, as it aims at minimizing the MSE. Fig. 4.5 shows image fragments from the original sequence Bicycle, the simulated one, the processed ones by the original method without and with these reduction techniques. These three reduction techniques degrade the performance of the proposed method without class reduction only little, while CS and EA show better performance at suppressing the ringing artifacts than OF.

(A) Original   (B) Simulated input   (C) Without reduction

(D) using CF   (E) using CS   (F) using EA
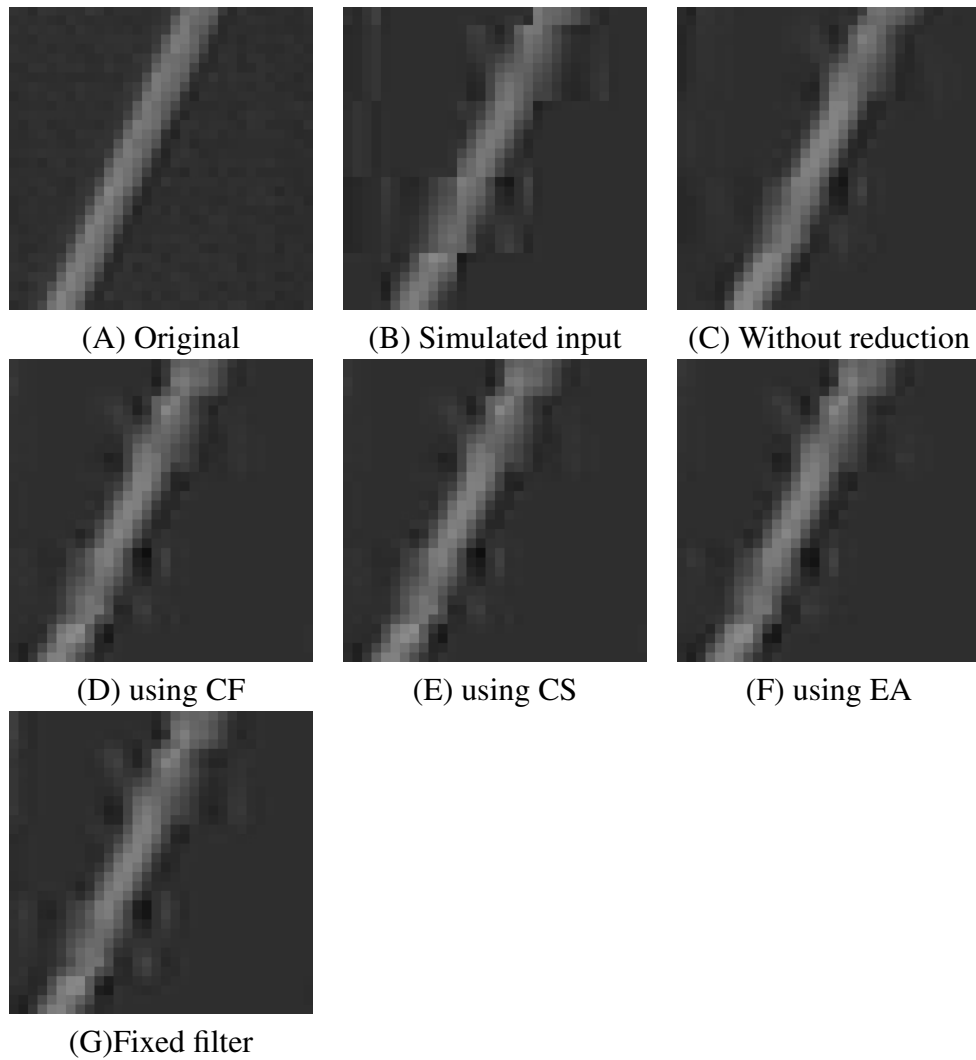
(G)Fixed filter

Figure 4.5: Image fragments from the results in the sequence Bicycle: (A) original, (B) simulated input, (C) without reduction, (D) using occurrence frequency, (E) using coefficient similarity , (F) using error advantage, (G) fixed filter.

Table 4.3: MSE scores of evaluated methods in coding artifact reduction.

	Mean Square Error				
Sequence	No reduction	CF	CS	EA	Fixed
Class No.	8192	32	32	32	1
Bicycle	76.9	88.0	87.0	85.1	97.7
Birds	15.7	17.0	16.5	16.9	19.8
Boat	87.5	91.1	90.7	90.2	96.5
Lena	40.4	43.0	42.6	42.3	46.0
Motor	141.6	150.2	149.1	148.0	157.5
Average	72.4	77.9	77.2	76.5	83.5

Table 4.4: MSE scores of evaluated methods in image interpolation.

	Mean Square Error					
Sequence	No reduction	CF	CS	EA	Fixed	Atkins'
Class No.	4096	100	100	100	1	100
Bicycle	41.4	47.0	45.6	45.2	65.4	49.8
Birds	20.3	20.4	20.3	20.2	22.4	22.1
Boat	55.1	57.7	56.9	56.9	63.2	59.3
Lena	78.1	82.6	81.5	81.4	90.2	83.1
Motor	52.4	53.1	52.8	52.7	53.1	52.7
Average	49.4	52.2	51.4	51.2	58.9	53.4

4.4.2 Application to image interpolation

For image interpolation, we apply these three class-count reduction techniques to Kondo's method. For the evaluation, we use some test sequences shown in Fig. 4.6, which are not included in the training set. The test sequences first are down-scaled two times to generate the down-scaled version as the simulated input. Then Atkins' method, original Kondo's method and the proposed class-count reduction methods are evaluated using these sequences. For a fair comparison, we use the same number of clusters for the proposed method and Atkins' method and the same aperture for the proposed methods and Kondo's method. Since Atkins' method performs best at the cluster number $M = 100$ [105], we use the same number here. Similar to the coding artifact reduction application, we also include a fixed filter for reference.

Table 4.4 shows the MSE results of the evaluated methods. One can see that

(A) Bicycle   (B) Lena

(C) Football   (D) Siena   (E) Tokyo
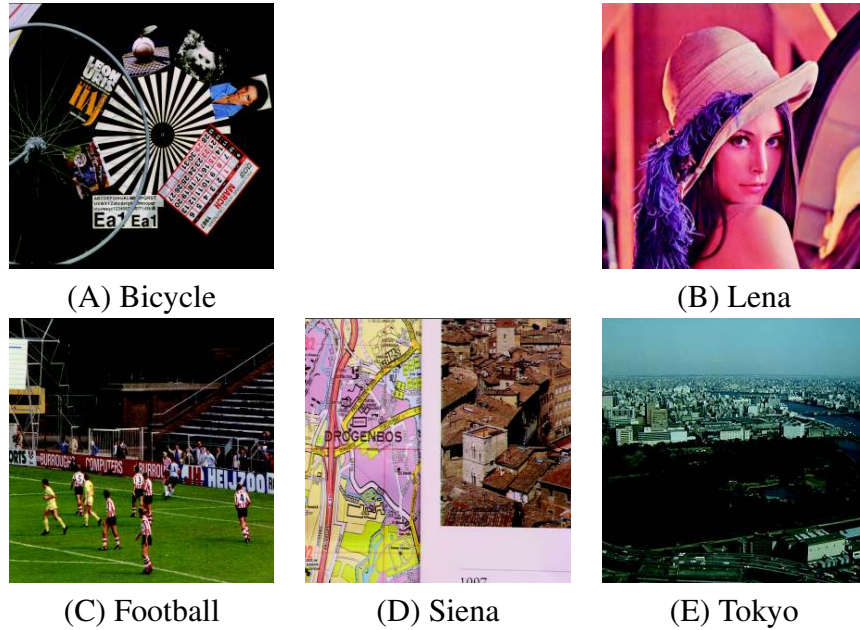
Figure 4.6: The testing material used for the evaluation in image interpolation.

the proposed method outperforms Atkins' method in terms of the MSE score, while it is far less computationally expensive for both the classification and obtaining the filter coefficients. Fig. 4.7 shows image fragments from the original high resolution sequence Bicycle and processed ones by all the methods. All these three proposed methods render the lines in different directions correctly where Atkins' method produces some staircase artifacts. Among them, EA and CS produce slightly smoother results at reconstructing the lines than CF. They show more or less the same interpolation quality as the original Kondo's method, though the LUT size had been reduced nearly by a factor of 40 and only one extra fetch operation is needed.

4.5   Conclusion

The proposed content adaptive processing concept offers a convenient and scalable video enhancement algorithm design at the possible cost of an exploding number of content classes. In this chapter, we have proposed three alternative class-count reduction techniques, class-occurrence frequency, coefficient similarity and error advantage for the proposed filtering framework. In the applications of coding artifact reduction and image interpolation, it has been shown that these techniques can greatly reduce the number of content classes without sacrificing
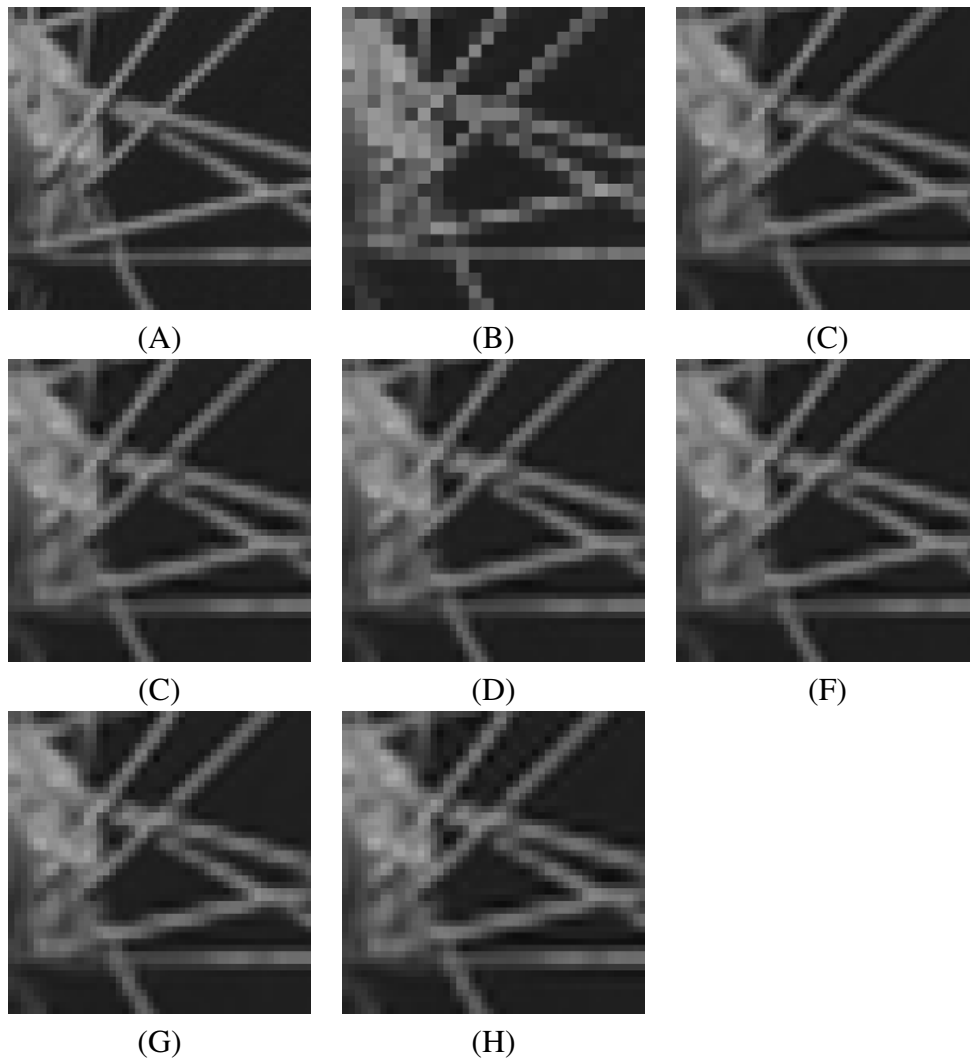
Figure 4.7: Image fragments from the results in the sequence Bicycle: (A) original, (B) downscaled version, (C) original Kondo's method, (D) using occurrence frequency, (E) using coefficient similarity , (F) using error advantage, (G) Atkins' method, (H) fixed filter.

much performance and are promising for applications using the proposed framework with a large number of features. Among them, the coefficient similarity and error advantage approach produce the best result. Taking the cost into consideration, the class-occurrence frequency approach seems to be the choice.

The reduction technique of using error advantage offers an MSE-optimal way to cluster or classify the content classes. This could be interesting for further research to find such an optimal classification on the raw pixel data for video enhancement. For instance, the classification on the input pixels still need to be classified first by using ADRC, which is not yet proven to be optimal, and a look-up-table is then used for the labeling, where a function that directly maps the input vector to the cluster number is more desirable. Finding such a function merits further attention.

Chapter 5

Nonlinear filtering

The proposed content adaptive processing framework includes two main parts, content classification and model selection. In the previous chapters, we have proposed new classifiers in the content classification part to extend the framework for more applications. So far in the thesis, the processing model in the framework has always been a linear filter. In this chapter, we try to answer the question what nonlinear filters could add to our results.

From the literature it is known that order statistics filters and bilateral filters may perform better in smoothing tasks where edge preservation is important [90][91][57]. Bilateral filters have the ability to locally adapt the filtering to the image content. It is unclear, however, how bilateral filters can be tuned to optimal adaptation given a filtering application like coding artifact reduction. Also it is interesting to see if the content classification, like the types we have used in the previous chapters, can still add to such an inherently adapting filter.

Like the bilateral filter and the linear filter, a neural network can be used as a neural filter which also inherently adapts to the local image content and already combines linear and nonlinear ingredients. It is, therefore, interesting to answer the question if the neural filter can profit from additional content classification and if it is the ultimate trained nonlinear filter, or not.

To answer the above questions in this chapter, we will study four different categories of nonlinear filters: order statistics filters, hybrid filters, neural filters, and bilateral filters with and without various forms of classification in different enhancement applications including image de-blocking, noise reduction, and image interpolation.

The chapter is organized as follows. We begin with an introduction about the nonlinear filters in Section 1. According to the way the nonlinear filters introduce nonlinearity, they can be classified into four categories. Four representative filters from these categories are reviewed and discussed in Section 2. Although these nonlinear filters are designed to adapt to the input content, there is no explicit

content adaption as we described in previous chapters. In order to investigate the additional performance improvements with the content adaption, we apply these filters in the proposed framework of content adaptive filtering in Section 3 and an intensive evaluation of these nonlinear filters in different video processing applications are provided in Section 4. Finally, we draw our conclusion in Section 5.

5.1   Introduction

The earliest and most widely used nonlinear filter probably is the median filter [85]. In the median filter, the median value in the filter window is the output of the filter. It shows good performance at removing impulsive noise and preserving edges [87]. In fact the median filter uses the order statistics information and the noisy values are regarded as outliers so that they can be removed. The further research about the median filter has led to a category of nonlinear filters which produce outputs based on the rank ordered observations, such as order statistic (OS) filters[90, 91]. Such filters based on only order statistics have some advantages over linear filters. They are robust in environments with impulsive interference and they can track signal discontinuities without introducing smooth transitions, as linear filters do. However, the rank order information alone is not sufficient in many applications. To incorporate both the spatial order and rank order information, many generalizations of rank-order filters have been proposed. Good examples among them are combination filters [92, 93], permutation filters [94, 95] and hybrid filters [96]. Different from the combination filters and the permutation filters which exhibit high complexity, the hybrid filter is relatively simple. The hybrid filter directly combines a linear filter and an OS filter. It exploits both the spatial and rank information in the image content and is proposed to realize the advantages of the OS filters in edge preservation and reduction of impulsive noise components while retaining the ability of the linear filter to suppress Gaussian noise.

With the introduction of the neural network to image processing, another type of nonlinear filters, the neural filter, has also been proposed [106, 108]. The neural filter is essentially a multi-layer feed forward neural network. The neural network takes the neighboring pixels from an image as the input and outputs the processed pixels. Rather than using the linear combination of the input pixel samples, a nonlinear transfer function at the hidden unit is applied to the weighted sum of the inputs. The flexibility of the neural network can be increased by using more hidden units or hidden layers. Because of its universal approximation property, the neural network can provide a better function approximation by a supervised learning. With the more flexible nonlinear model, the neural filter has shown

better performance than the linear filter [89].

The third category of nonlinear filters includes edge-preserving smoothing methods which utilize pixel similarity information. The early approaches including the sigma filter [86] and the fuzzy filter [99], give the weights of input pixels according to their value differences from the central pixel value. More recently the bilateral filter [57] has received considerable attention in areas of image processing and computer vision. Unlike the sigma filter and the fuzzy filter of which the coefficients are determined by the pixel value difference, the bilateral filter adjusts its coefficients to the spatial closeness and photometric similarity of the pixels. Due to this adaptivity, it has shown good performance at edge-preserving smoothing for image processing applications, such as noise reduction and digital coding artifact reduction [59]. For a linear filter, its coefficients can be adjusted to achieve desired effects by a supervised learning and the least mean square optimization. However, this is not trivial for the bilateral filter. In order to solve that problem, we proposed a new type of filter, the trained bilateral filter [130]. The trained bilateral filter adopts a linear combination of spatially ordered and rank ordered pixel samples, which has been proposed in a hybrid filter. Different from the hybrid filter where the similarity had been heavily quantized, the rank ordered pixel samples in the proposed method are further transformed to reflect the photometric similarity of the pixels. Consequently, the trained bilateral filter possesses the essential characteristics of the original bilateral filter. On the other hand, the design of the proposed bilateral filter makes it feasible to optimize the filter coefficients. That is, the optimal coefficients for the combined pixel samples can be obtained by the least mean square optimization as for the linear filters.

5.2 Nonlinear filters

In this section, we choose four representative nonlinear filters from the mentioned categories in the previous section, the order statistics filter which only uses the rank order information, the hybrid filter which combines the rank order and spatial information, the trained bilateral filter which adopts both the spatial and similarity information and the neural filter which introduces the nonlinear transfer function. The definitions and properties of these filters are then reviewed.

5.2.1 Order statistic filter and hybrid filter

Linear filters estimate the output by using a weighted sum of the observation samples in the spatial order. They have good performance at eliminating Gaussian noise, but they also blur the signal edge [88]. Order statistics filters that are based on rank order information have been introduced to solve the problem.

Such filters track signal discontinuities so that they can provide better edge preserving. However, using the rank order information alone has a limitation in many applications. The limitation can be explained by a simple example. Consider the observation vectors $X_1 = (181, 183, 182, 85, 77, 76, 180, 185, 190)$ and $X_2 = (181, 183, 182, 180, 185, 190, 85, 77, 76)$, which are observations in a 3×3 aperture of a line and an edge in an image, respectively. Although these vectors have very distinct and different patterns, their corresponding sorted vectors are identical: $X_r = (76, 77, 85, 180, 181, 182, 183, 185, 190)$. Clearly, rank order based filters fail to exploit the spatial context within the filter aperture. To incorporate both the spatial order information and the rank order information, the hybrid filter is proposed to combine a linear filter and an OS filter so that it can realize the advantages of both filters.

Let us start with the definition of a linear filter. Let $X = (x_1, x_2, ..., x_n)^T$ be an observation containing n samples arranged by the spatial or temporal order in which the samples are observed. X_r is the sorted observation vector $X_r = (x_{(1)}, x_{(2)}, ..., x_{(n)})^T$ where $x_{(i)}$ is the ith largest sample in X, so that $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$. Let the observation vector X be the input to the filter. For the linear filter, we have

$$y = W^T X \tag{5.1}$$

where y is the output of the linear filter and W is an $N \times 1$ vector of coefficients for the linear filter. Consequently, the linear filter only takes consideration of the spatial position of the pixel samples.

Then for an OS filter, we have

$$y_r = W_r^T X_r \tag{5.2}$$

where y_r is the output of the OS filter and W_r is an $N \times 1$ vector of coefficients for the OS filter.

By concatenating X and X_r we can obtain an extended vector X_h which contains spatial ordered and rank ordered samples.

$$X_h = (x_1, x_2, ..., x_n, x_{(1)}, x_{(2)}, ..., x_{(n)})^T \tag{5.3}$$

The hybrid filter is a linear combination of both spatial ordered and rank ordered samples as shown in Equation 5.4.

$$y_h = W_h^T X_h \tag{5.4}$$

where y_h is the output of the hybrid filter and W_h is a $2N \times 1$ vector of coefficients for the hybrid filter.

As one can see from Equation 5.4, if the coefficients for the spatial ordered samples or the rank ordered samples are constrained to be zero, the hybrid filter becomes equal to the OS filter or the linear filter, respectively.

The optimization of the hybrid filter can be accomplished in a similar fashion as for the linear filter. Suppose the output of the hybrid filter $y_h(t) = W_h^T X_h(t)$ is used to estimate the desired signal $d(t)$. The optimal filter coefficients are obtained when the mean square error between the output and desired signal is minimized. The mean square error MSE is:

$$MSE = E[(y_h(t) - d(t))^2] = E[(W_h^T X_h(t) - d(t))^2]. \qquad (5.5)$$

Taking the first derivative with respect to the weights and setting it to zero, we obtain [127]:

$$W_h^T = E[X_h X_h^T]^{-1} E[X_h d]. \qquad (5.6)$$

5.2.2 Trained bilateral filter

The trained bilateral filter is inspired by the bilateral filter and the hybrid filter. The bilateral filter is proposed as a generalization of other edge-preserving smoothing filters such as the sigma filter [86] and the fuzzy filter [99]. It adapts its coefficients to the spatial closeness and photometric similarity of the pixels. Consequently it shows very good performance at edge-preserving smoothing. The output y_b of a bilateral filter is defined by [57]:

$$y_b = \frac{\displaystyle\sum_{i=1}^{N} x_i \cdot c(x_i, x_c) \cdot s(x_i, x_c)}{\displaystyle\sum_{i=1}^{N} c(x_i, x_c) \cdot s(x_i, x_c)}$$

$$s(x_i, x_c) = exp[-(x_c - x_i)^2 / 2\sigma_s^2]$$
$$c(x_i, x_c) = exp[-d(x_c, x_i)^2 / 2\sigma_c^2]$$
$$i = 1, 2, ..., N. \qquad (5.7)$$

where x_c is the spatially central pixel and $d(x_c, x_i)$ is the Euclidean distance between the pixel position of x_i and x_c. The Gaussian function has been typically used to relate coefficients to the geometric closeness and photometric similarity of the pixels, which seems somewhat arbitrary. Also it is not obvious how to optimize the bilateral filter using a supervised learning like the hybrid filter.

As one can see, the hybrid filter incorporates both the rank order and spatial position information as the bilateral filter. However, the rank ordering in the hybrid filter only gives some indications of the pixel similarity, that is, the similarity has been heavily quantized. In order to incorporate the complete similarity information as the original bilateral filter does, we obtain the vector $X_s = (x_{[1]}, x_{[2]},$

$..., x_{[N]})^T$ by sorting the pixels in the filter aperture according to their pixel value distance to the spatially central pixel x_c. The ordering is defined by:

$$|x_{[i+1]} - x_c| \geq |x_{[i]} - x_c|, i = 1, 2, ..., N. \tag{5.8}$$

Then we transform the vector X_s into $X_s\prime = (x_{[1]}\prime, x_{[2]}\prime, ..., x_{[N]}\prime)^T$. The transform is defined as:

$$x_{[i]}\prime = \mu(x_c, x_{[i]}) \cdot x_c + (1 - \mu(x_c, x_{[i]})) \cdot x_{[i]}, i = 1, 2, ..., N. \tag{5.9}$$

where $\mu(x_c, x_{[i]})$ is a membership function between $x_{[i]}$ and x_c. The membership function is defined as:

$$\mu(x_c, x_{[i]}) = MIN(\frac{|x_{[i]} - x_c|}{K}, 1). \tag{5.10}$$

where K is a pre-set constant. Other membership functions such as a Gaussian function are also possible. The vector X_{tb} is obtained by concatenating the vectors X and $X_s\prime$:

$$X_{tb} = (x_1, x_2, ..., x_n, x_{[1]}\prime, x_{[2]}\prime, ..., x_{[N]}\prime)^T. \tag{5.11}$$

Similar to the linear filter, we define the output of the proposed trained bilateral filter as:

$$y_{tb} = W_{tb}^T X_{tb}. \tag{5.12}$$

where W_{tb} is a $2N \times 1$ vector of weights.

The expected advantage of the trained bilateral filter is that, the weights of the transformed samples that are similar to the center sample value are increased to better preserve edges and suppress the noise. On the other hand, the linear part obtains the spatial information which is useful for local image structure reconstruction. Essentially the trained bilateral filter behaves as the original bilateral filter whose coefficients are continuously dependent on the spatial and intensity difference of pixels. Additionally the coefficients of the trained bilateral filter can be optimized by a supervised learning in a similar fashion as the hybrid filter.

5.2.3   Neural filter

Different from other filters that use the rank order and similarity information, the neural filter introduces the nonlinearity by using a nonlinear transfer function. In the neural filter, a multi-layer feed-forward neural network is employed as a convolution kernel. The neural network takes the pixels in a filter window from the input image and outputs the processed pixel as the result of the neural network computation. A two-layer neural network with N_h hidden units as shown in Fig. 5.1 is defined by:

$$y_{nn} = f_2(LW f_1(IWX + b_1) + b_2). \tag{5.13}$$
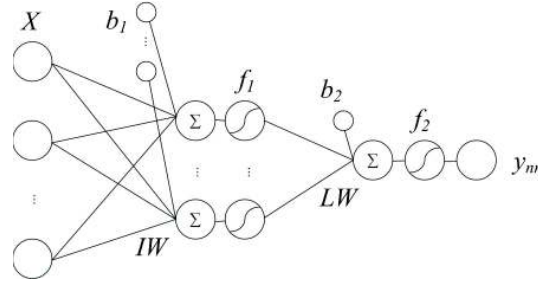
Figure 5.1: The two-layer neural network model with several hidden units at the hidden layer.

where IW is an $N_h \times N$ matrix of weights connecting the input layer to the hidden layer; LW is a $1 \times N_h$ matrix of weights for the hidden layer; b_1 is an $Nh \times 1$ matrix of bias for the hidden layer; b_1 is a bias for the output and f_1, f_2 are transfer functions for the hidden and output layer, respectively. The transfer function can be an identity function or a sigmoid function. Functions such as the hyperbolic tangent that produce both positive and negative values are usually chosen for the hidden layer. Such functions tend to yield a faster training than functions that produce only positive values such as log-sigmoid, because of better numerical conditioning [111]. The identity function is often employed in the output layer because the characteristics of a neural network are improved significantly with an identity function when applied to function approximation issues in image processing [76]. When all the transfer functions are identity functions, the neural filter becomes a linear filter. The flexibility of the neural network can be increased by using more hidden units or hidden layers.

The neural network acquires various nonlinear functions by a supervised learning. The optimal coefficients for a neural network can be obtained through back-propagation [107]. During the training, the errors between outputs and targets are computed and the derivatives of the errors are back-propagated to adjust the coefficients of the network iteratively and minimize the mean squared errors.

5.3 Content adaption

As one can see in the previous section, these nonlinear filters do not explicitly utilize the content classification from which the linear filter can profit a lot as shown in the previous chapters. We expect the content classification could bring additional performance improvement to these nonlinear filters. Therefore we apply the nonlinear filters in the proposed content adaptive filtering framework.

To apply the nonlinear filters in the proposed framework, we simply replace the linear filtering part with a nonlinear filter as shown in Fig. 5.2. The training

procedure is similar to the one for the linear filter. We optimize the coefficients of these nonlinear filters for every class.
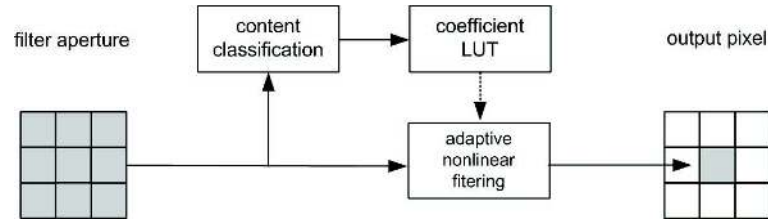
Figure 5.2: The block diagram of using nonlinear filters in the proposed framework: the local image structure is classified using the content classification and the filter coefficients are obtained from the LUT.

5.4 Experiments and results

In this section, an evaluation of the four mentioned nonlinear filters in different image processing applications, image de-blocking, noise reduction and image interpolation, is provided. In the evaluation, we compare these filters with the linear filter. And also different content classifications are investigated.

Training and test material

The training material includes a variety of high quality natural images, including people, building, animals and landscapes. All the filters are trained on the same training material. And the test images and the snapshots from the test sequences used in our experiments are shown in Fig. 5.3. Note that the test material is not included in the training material.

Filter setting

For the neural filter, a two layer feed forward neural network is used. The transfer function used in the hidden layer is the hyperbolic tangent function, whereas the identity function is used at the output layer. The pixel value range in the neural filter is re-scaled from the range [0, 255] to [-1, 1], which corresponds to the output range of the hyperbolic tangent function. For a fair comparison, we use two hidden units in the hidden layer, which will result in a similar number of coefficients as the hybrid filter and the trained bilateral filter. Table 5.1 lists the numbers of the coefficients of different filters per class.

(A) Bicycle   (B) Birds   (C) Boat

(D) Football   (E) Lena   (F) Motor
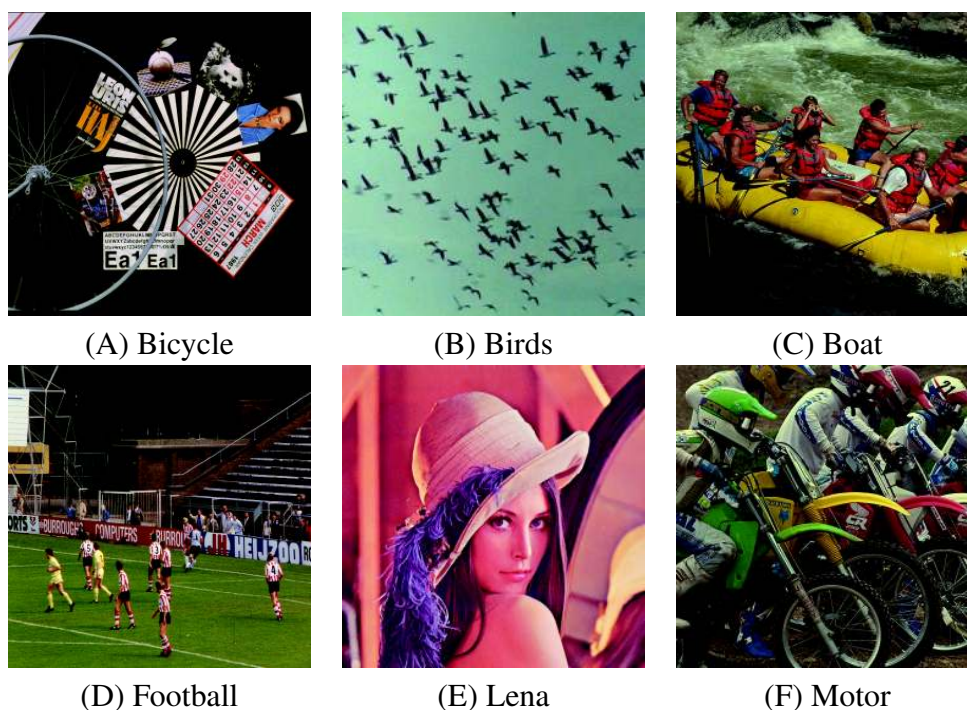
Figure 5.3: The testing material used for the evaluation.

Table 5.1: Number of coefficients of different filters per class.

Filter	Linear	OS	Hybrid	Trained bilateral	Neural
Coefficient number	N	N	2N	2N	2N+5

Evaluation procedure

For the evaluation, we degrade (compress, add noise, down-scale) the original test sequence to generate the simulated input sequences. Then different filters are applied to the simulated input sequences. The MSE scores between the original test sequences and processed sequences will be used as the performance indicator.

5.4.1   Image de-blocking

In the experiment for image de-blocking, we evaluate the filter performance to remove JPEG compression coding artifacts. The test images and sequences have been compressed using JPEG compression at a quality factor of 20 (the quality factor of 100 is the best). The free baseline JPEG software from the Independent

JPEG Group website [1] is used for the JPEG encoding and decoding. We use a diamond shape filter window shown in Fig. 5.4 to balance between the performance and the complexity. As suggested in Chapter 2, we use the ADRC classification and the DR classification for the content classification. To show the contribution from the individual classifiers, we separately investigate the ADRC classification, the DR classification and their combination. For the DR classification $Tr = 32$ is used.
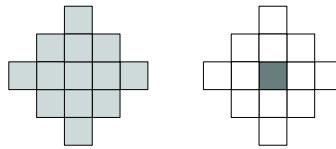
Figure 5.4: The diamond shape filter window for de-blocking: the estimated output is in the center of the window.

In addition to the MSE score, we also use the BIM metric proposed by Wu [101] for the evaluation. The BIM metric measures the blockiness of compressed images or sequences. The BIM value $BIM = 1$ refers to no blockiness at all and the larger the BIM value is, the more blockiness in the content. A lower BIM value can be achieved by a strong smoothing filter. However, this will remove lots of details and increase the MSE score. Therefore, we use both the MSE and BIM scores for the evaluation.

The MSE scores of all the filters with different classifiers are shown in Table 5.2. The average BIM scores of the test sequences processed by these filters are shown in Fig. 5.5. For the MSE score, all the nonlinear filter, except the OS filter, perform better than the linear filter. For the BIM score, all the nonlinear filters have better results than the linear filter, while the OS filter has the lowest BIM score. For both the MSE and BIM scores, all the filters can benefit from the ADRC classification and the DR classification. With the combination of the ADRC and DR classification, the best MSE and BIM scores are achieved. The OS filter has the highest MSE score because it only uses the rank order information and fails to exploit the structure information. This can be shown in the MSE scores for the sequences such as Bicycle, Boat and Motor which contain many image details. With the ADRC classification, the performance of the OS filter can be greatly improved, although it is still worse than the linear filter. The hybrid filter has shown better performance than the linear filter due to the added rank order information. With a similar complexity as the hybrid filter, the trained bilateral filter shows much better performance in the MSE score, even without the content

---

[1] The web address is: http://www.ijg.org/files/jpegsrc.v6b.tar.gz

Table 5.2: MSE scores for de-blocking.

	Mean Square Error							
Sequence	Linear				OS			
Classification*	I	II	III	IV	I	II	III	IV
Bicycle	54.1	53.7	45.8	45.3	89.1	89.0	63.0	62.4
Birds	11.6	11.6	11.3	11.8	17.2	17.3	14.6	13.6
Boat	64.4	64.3	62.5	61.6	87.7	87.5	73.2	72.5
Lena	31.9	31.9	30.2	30.1	39.4	39.3	34.0	33.3
Motor	96.7	96.3	89.2	89.3	156.6	156.2	116.4	115.2
Average	51.7	51.5	47.8	47.6	78.0	77.9	60.2	59.4

	Mean Square Error							
Sequence	Hybrid				Tr-bilateral			
Classification*	I	II	III	IV	I	II	III	IV
Bicycle	50.2	49.8	48.5	44.7	45.0	44.9	41.5	41.5
Birds	11.5	11.5	11.0	10.9	10.9	10.9	10.5	10.5
Boat	63.1	62.9	62.3	61.2	60.0	59.9	59.5	59.5
Lena	31.1	31.1	30.2	29.6	30.0	29.9	29.0	28.8
Motor	93.7	93.1	90.3	88.5	86.9	86.8	84.2	84.2
Average	49.9	49.6	48.5	47.0	46.6	46.4	45.1	45.1

	Mean Square Error				
Sequence	Neural				Compressed
Classification*	I	II	III	IV	
Bicycle	53.7	51.9	45.9	45.3	63.5
Birds	11.2	11.2	11.2	11.2	13.8
Boat	63.4	62.8	61.4	61.8	72.3
Lena	31.9	31.0	29.5	29.3	36.9
Motor	95.4	94.3	89.1	89.1	108.9
Average	51.1	50.2	47.4	47.3	59.1

*Classification: I - no classification, 1 class, II - DR, 2 classes, III - ADRC, 4096 classes, IV - ADRC+DR, 8192 classes.
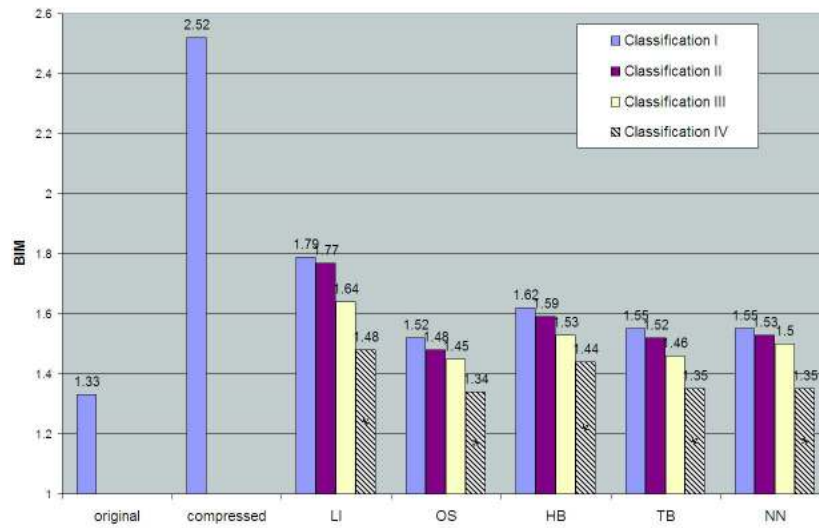
Figure 5.5: The average BIM scores of the test sequences processed by these filters with different classifications: LI-Linear filter, OS-Order statistics filter, HB-Hybrid filter, TB-Trained bilateral filter, NN-Neural filter. Classification: I - no classification, 1 class, II - DR, 2 classes, III - ADRC, 4096 classes, IV - ADRC+DR, 8192 classes.

classification. And it also achieves a relatively low BIM score. This suggests that the trained bilateral filter has a stronger signal adaptivity when the similarity information is incorporated so that the additional content classifications will not bring much improvement.

This is also reflected in the image fragments from Motor image processed by all the filters shown in Fig. 5.6-5.7. With the classification the linear filter can suppress the coding artifact nicely, but the edges are also blurred comparing to the original. The OS filter can greatly reduce the coding artifacts but it also destroys all the fine structural details. Although it shows better details preserving with the structure classification, the overall performance is not still as good as the linear filter. Comparing to the linear filter, both the hybrid filter and the neural filter can equivalently suppress the coding artifacts and demonstrate a better ability at preserving edges. As suggested in the MSE and BIM evaluation, the trained bilateral filter demonstrates the best edge preserving ability and removes coding artifacts effectively. When comparing the results of using different classifications, we see that using the ADRC classification improves the performance at the fine details and using the ADRC+DR classification removes the blocking artifacts in the flat area better than using the ADRC classification alone.

(A) Original   (C) LI   (G) OS

(B) Corrupted   (D) LI DR   (H) OS DR

(E) LI ADRC   (I) OS ADRC

(F) LI ADRC+DR   (J) OS ADRC+DR

Figure 5.6: Image fragments from the image Motor processed by different filters with different classifications: LI-Linear filter, OS-Order statistics filter.

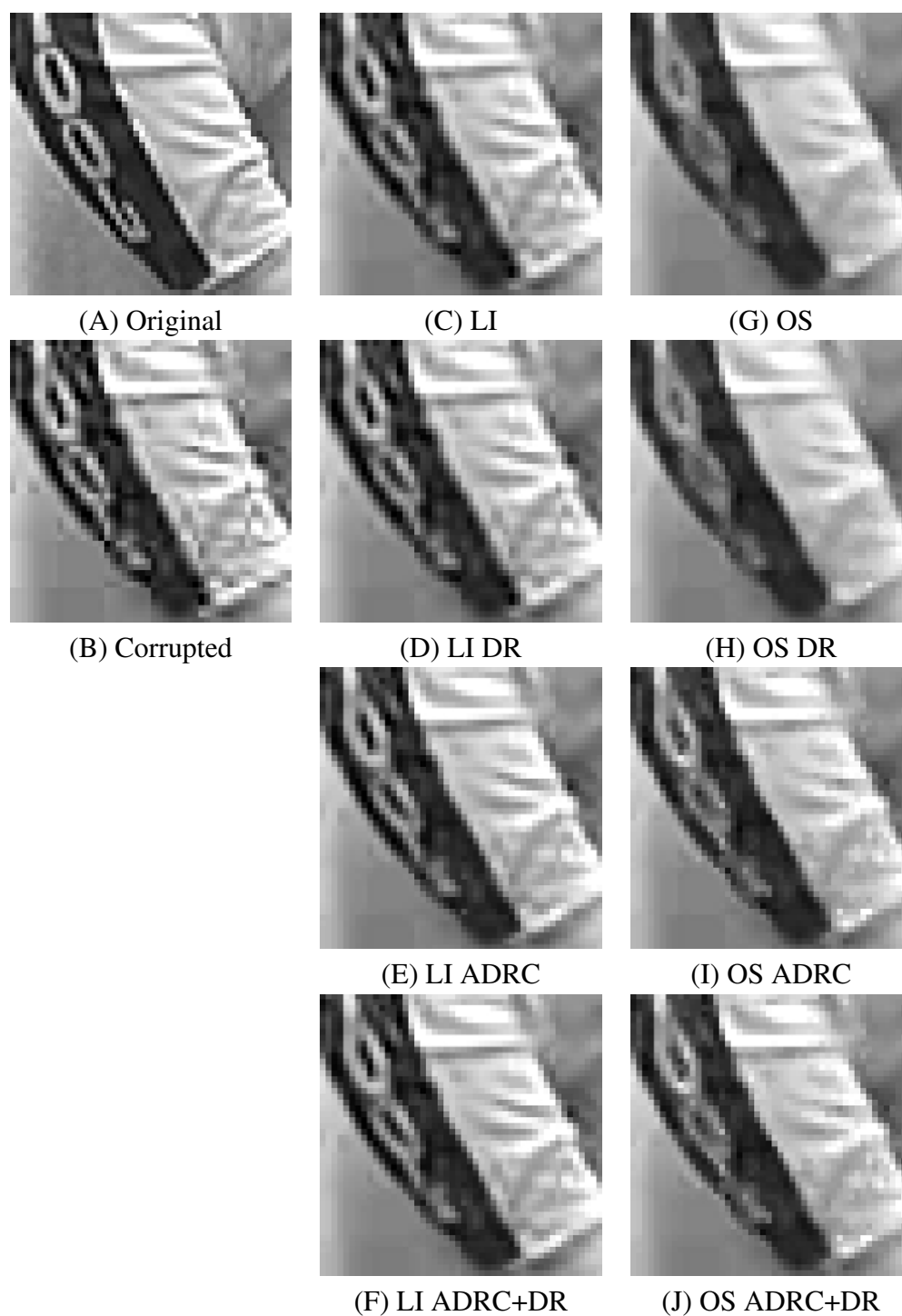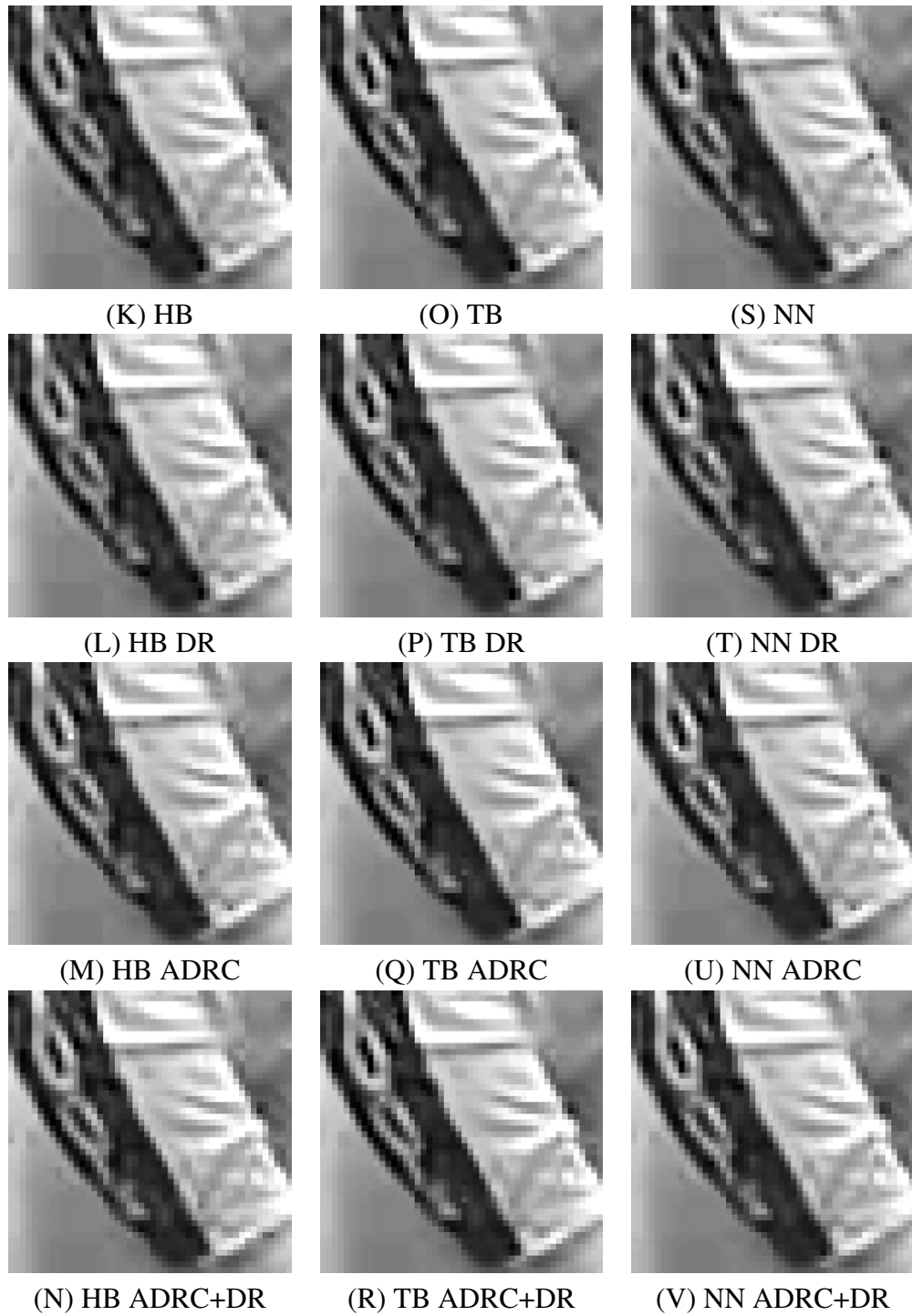| (K) HB	(O) TB	(S) NN
(L) HB DR	(P) TB DR	(T) NN DR
(M) HB ADRC	(Q) TB ADRC	(U) NN ADRC
(N) HB ADRC+DR	(R) TB ADRC+DR	(V) NN ADRC+DR

Figure 5.7: Image fragments from the image Motor processed by different filters with different classifications: HB-Hybrid filter, TB-Trained bilateral filter, NN-Neural filter.

5.4.2 Noise reduction

For noise reduction, we will evaluate these filters' abilities to remove Gaussian noise. The Gaussian noise usually manifests itself as irregular luminance patterns, which are different from real image structures. We expect that the ADRC classification could help distinguish the noise from the real image structures so that better noise reduction can be achieved. And also we hope that better noise reduction can be achieved in the low contrast area, in which case the DR classification is needed. Therefore, in the experiment, the content classifications, ADRC, DR and ADRC+DR, are investigated. The Gaussian noise applied here has a mean of 0 and a standard deviation of 10. A 3×3 filter window centered at the pixel to be estimated is employed to eliminate the noise. The threshold used in the DR classification is optimized to $Tr = 40$.

Table 5.3 lists the MSE scores of all the methods in the applications of Gaussian noise reduction. Similar to the results of image de-blocking, all the filters can benefit from the ADRC classification and the DR classification. With the combination of the ADRC and DR classification, the best MSE scores are achieved. Although the OS filter produces the worst MSE, with the content classification it still has a quite close score to the linear filter. It suggests that the rank order information has some effect at removing the noise. This is also shown in the results of the hybrid filter. The MSE score of the hybrid filter is improved by combining the linear filter and the OS filter. The trained bilateral filter has a significant improvement over the hybrid filter, given the fact that they have a similar complexity. Without any content classification, the trained bilateral filter achieves a better MSE score than any other filter with the content classification.

To enable a qualitative comparison, some image fragments from the sequence Bicycle restored by all the filters are shown in Fig. 5.8 - 5.9. The OS filter shows a strong noise reduction, but it also removes the details. Although the edge preserving performance of the OS filter can be further improved by the content classification, it is still not as good as the other nonlinear filters. The hybrid filter shows better performance at preserving edges than the linear filter. The trained bilateral filter further improves the edge preserving, producing the best contrast. The neural filter shows similar edge preserving, but it also produces some overshoots near the edges. When comparing the results of using different classifications, we see that using the DR classification improves the contrast a little and using the ADRC classification improves the performance at reconstructing the fine details. Furthermore, we see that the trained bilateral filter shows a great flexibility. It has good performance no matter whether the content classification is included. Without the content classification, it shows better performance than the linear filter with the content classification.

From the results of image de-blocking and noise reduction, we can conclude

Table 5.3: MSE scores for noise reduction.

	Mean Square Error							
Sequence	Linear				OS			
Classification*	I	II	III	IV	I	II	III	IV
Bicycle	57.1	52.4	40.1	37.7	68.1	65.3	46.3	43.2
Birds	25.3	22.7	19.8	19.2	21.6	21.0	20.2	20.0
Boat	54.1	51.9	46.3	45.2	73.4	72.4	53.0	51.5
Lena	33.0	31.3	28.5	28.0	37.9	37.1	30.5	30.2
Motor	79.2	74.7	60.5	58.1	114.7	112.0	72.0	67.9
Average	49.7	46.6	39.0	37.6	63.1	61.6	44.4	42.6

	Mean Square Error							
Sequence	Hybrid				Tr-bilateral			
Classification*	I	II	III	IV	I	II	III	IV
Bicycle	48.8	43.6	39.6	36.6	32.0	31.4	29.4	28.8
Birds	24.3	22.8	19.7	19.3	20.2	19.8	19.0	18.8
Boat	52.3	49.2	46.3	44.3	39.8	39.9	38.8	38.5
Lena	31.6	30.7	28.4	28.0	28.0	27.7	26.9	26.8
Motor	73.6	66.6	60.2	55.7	47.9	47.8	45.3	44.6
Average	46.1	42.5	38.8	36.7	33.6	33.3	31.9	31.5

	Mean Square Error				
Sequence	Neural				Corrupted
Classification*	I	II	III	IV	
Bicycle	49.7	47.6	33.1	31.8	100.3
Birds	20.6	20.6	19.4	20.1	100.4
Boat	48.3	48.0	42.6	41.9	100.2
Lena	31.6	31.3	27.6	27.8	100.3
Motor	73.0	71.0	56.8	54.7	100.2
Average	44.6	43.7	35.9	35.3	100.3

*Classification: I - no classification, 1 class, II - DR, 2 classes, III - ADRC, 4096 classes, IV - ADRC+DR, 8192 classes.

(A) Original

(C) LI

(G) OS

(B) Corrupted

(D) LI DR

(H) OS DR

(E) LI ADRC

(I) OS ADRC

(F) LI ADRC+DR

(J) OS ADRC+DR

Figure 5.8: Image fragments from the sequence Bicycle processed by different filters with different classifications: LI-Linear filter, OS-Order statistics filter.

(K) HB   (O) TB   (S) NN

(L) HB DR   (P) TB DR   (T) NN DR

(M) HB ADRC   (Q) TB ADRC   (U) NN ADRC

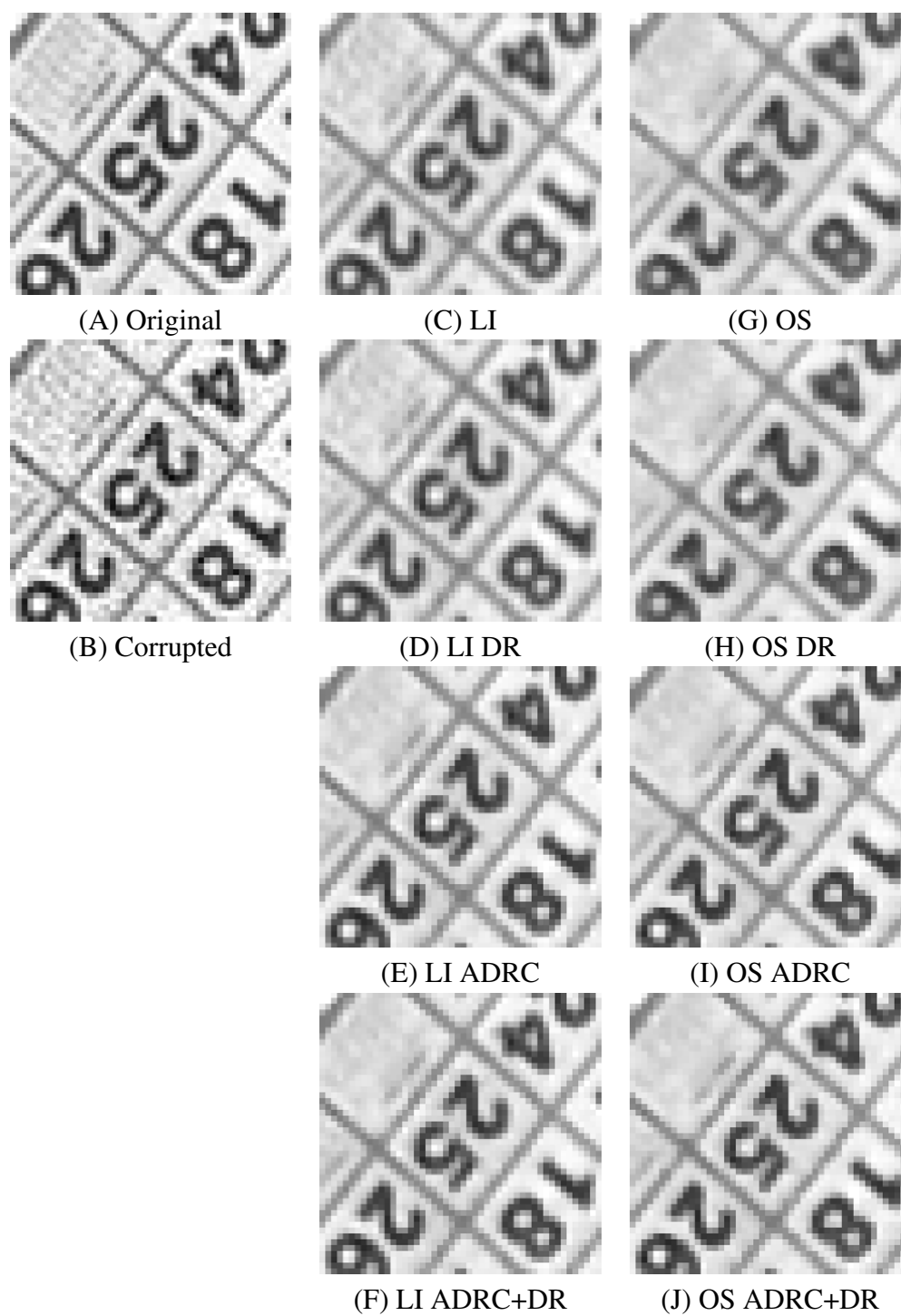(N) HB ADRC+DR   (R) TB ADRC+DR   (V) NN ADRC+DR
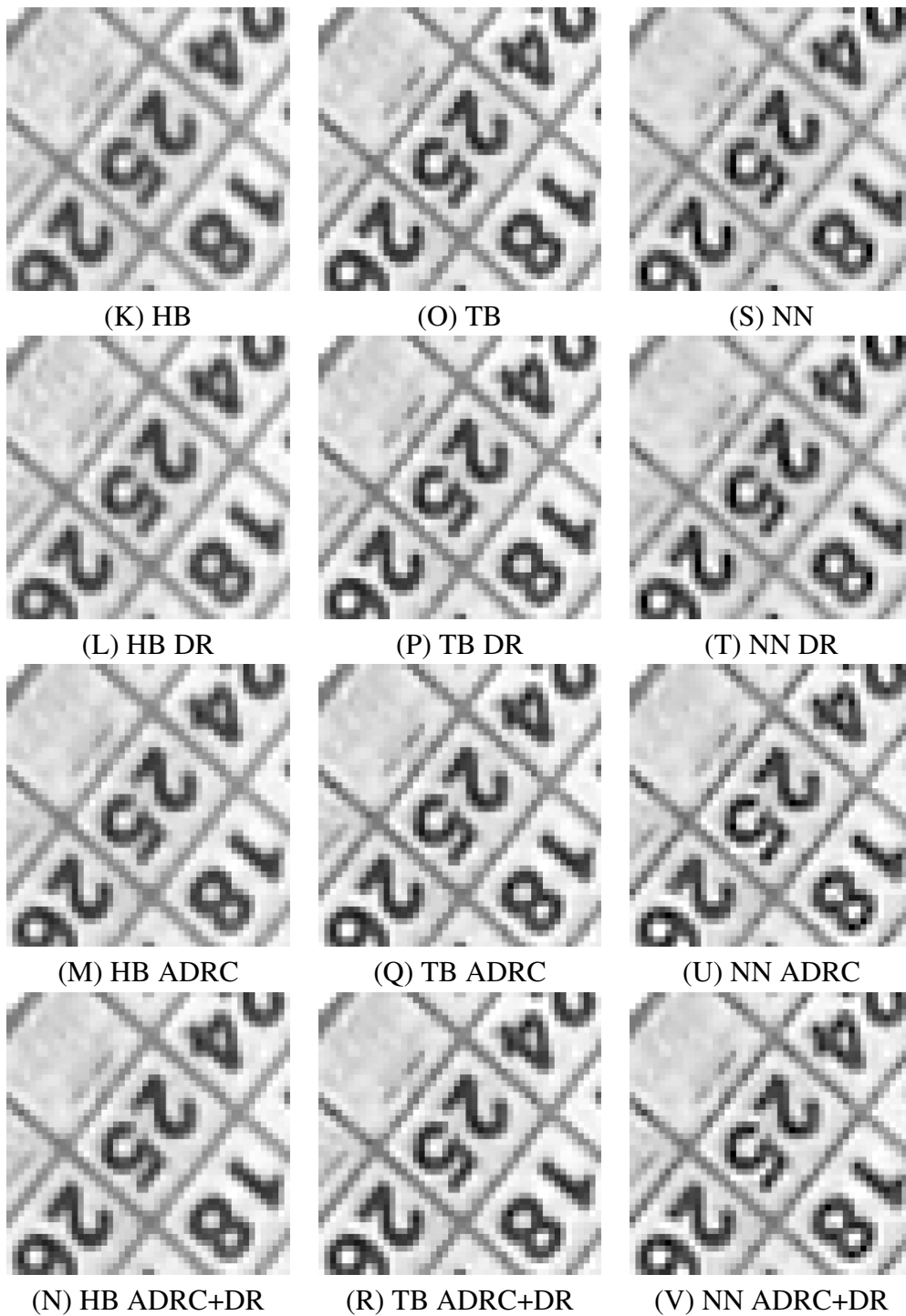
Figure 5.9:   Image fragments from the sequence Bicycle processed by different filters with
different classifications: HB-Hybrid filter, TB-Trained bilateral filter, NN-Neural filter.

that similarity information is very useful for noise reduction applications. The rank information only gives some indication about the pixel similarity, therefore the hybrid filter profits little from it. Although the neural filter is regarded as a flexible model which can approximate any smooth function, it still heavily depends on the content classification to get satisfactory results. From the results, we can see that a filter that is designed to inherently adapt to signal can achieve similar performance as a non-inherently adaptive filter that is based on the content classification. It also suggests that the performance of the neural filter can be further improved by inherent adaptations like the trained bilateral filter.

5.4.3 Image interpolation

In image interpolation, local structure classification has proven to bring significant improvement for the linear filtering [110]. We expect that the nonlinear filter based on the structure classification can further be improved. Because interpolation does not change with the local contrast, we use only ADRC for the content classification. In the experiment, we apply all the filters with an aperture size of 3×3 on the low resolution pixels to estimate the corresponding high resolution pixels using window flipping [98]. We adopt the same evaluation process as Zhao [110].

In Table 5.4, the MSE scores on the test images and sequence in the application of image interpolation are provided. The table shows that the OS filter has the highest MSE score because they only use the rank order information and fail to exploit the content structure. The MSE scores for these filters with the ADRC classification have a significant reduction compared to those without on every test image and sequence, which suggests that the structure information is important for interpolation. Comparing the results from the linear filter, the hybrid filter and the trained bilateral filter, we see that the rank order information and the similarity information do not bring much improvement as they do not contribute to better interpolation. The neural filter demonstrates a somewhat more robust estimation and achieves the lowest MSE score.

For a qualitative comparison, some image fragments from the Bicycle sequence interpolated by these methods are shown in Fig. 5.10. Without the ADRC classification, none of these filters produces satisfactory results, especially the OS filter destroys the local structure heavily. With the ADRC classification, more image details have been reconstructed due to the local structure information. The results from the linear filter and the hybrid filter show that they generate some staircase effects at some lines, while those lines are reconstructed more smoothly by the trained bilateral filter and the neural filter. Comparing the results from the trained bilateral filter and the neural filter, we can also see that the neural filter reproduces thinner lines that are closer to the original.

Table 5.4: MSE scores for image interpolation.

Sequence	Mean Square Error					
	Linear		OS		Hybrid	
Classification*	I	III	I	III	I	III
Bicycle	68.9	46.6	294.8	80.1	69.1	45.7
Birds	62.3	54.9	179.6	80.6	62.3	54.7
Boat	65.4	58.1	271.1	87.9	65.6	59.4
Lena	23.6	20.9	94.2	33.4	23.6	21.1
Motor	109.8	88.1	334.5	138.1	109.9	86.7
Average	66.0	53.7	234.8	84.0	66.1	53.5

Sequence	Mean Square Error			
	Tr-bilateral		Neural	
Classification*	I	III	I	III
Bicycle	69.0	44.4	63.0	43.2
Birds	62.2	54.4	61.3	54.6
Boat	65.3	57.3	64.0	55.9
Lena	23.5	20.5	23.2	20.0
Motor	109.8	89.1	104.2	87.6
Average	66.0	53.1	63.1	52.3

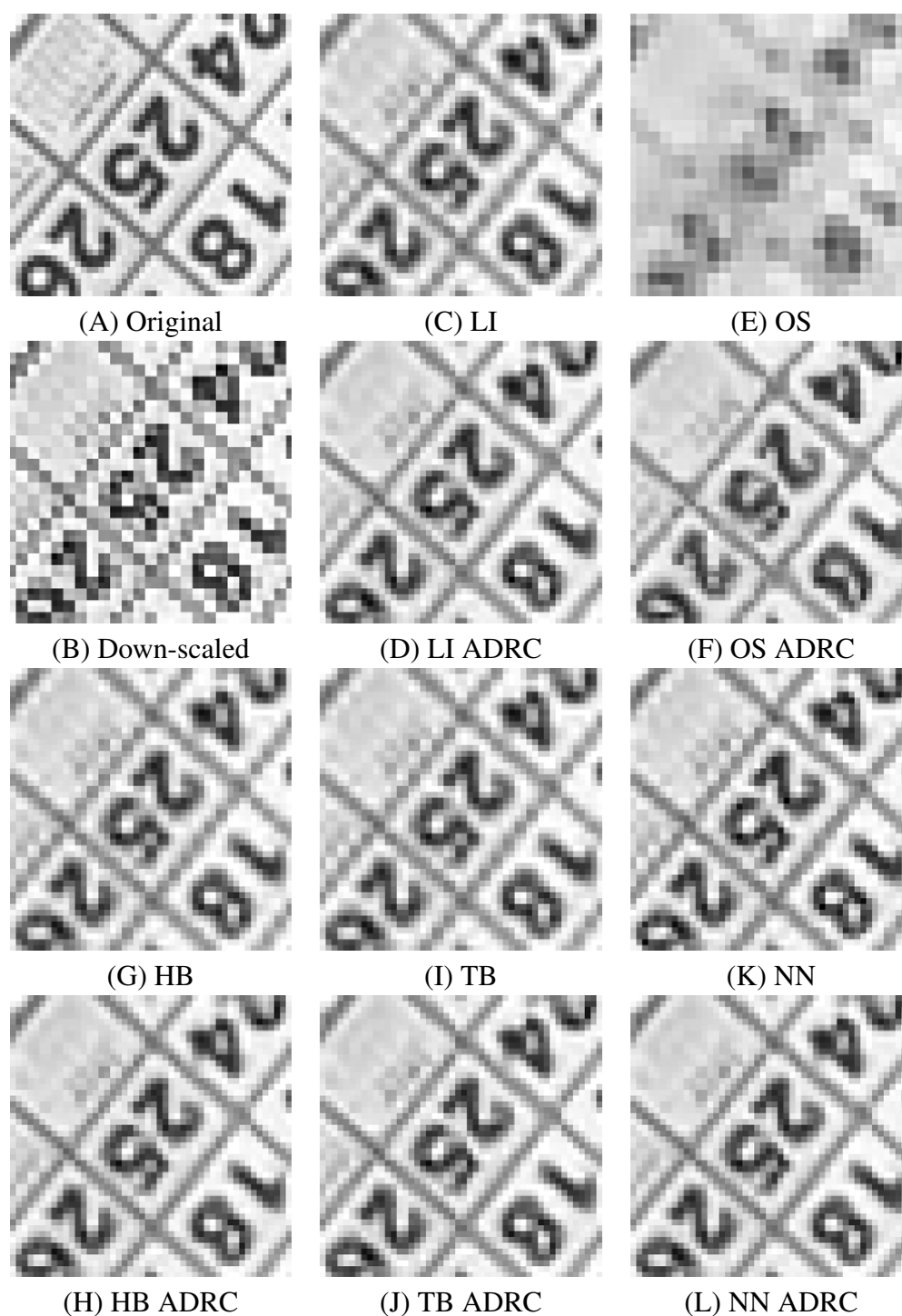*Classification: I - no classification, 1 class, III - ADRC, 256 classes

Figure 5.10: Image fragments from the Bicycle sequence interpolated by different filters with different classifications: LI-Linear filter, OS-Order statistics filter, HB-Hybrid filter, TB-Trained bilateral filter, NN-Neural filter.

5.4.4   Nonlinearity analysis

From the results in the previous section, we see that these nonlinear filters can bring performance improvement over the linear filter. To further gain insight of the nonlinear filter, we perform an analysis to see where the nonlinearity has been used. Here we choose the trained bilateral filter and the neural filter.

For the trained bilateral filter, the input vector X_{tb} consists of vector X and $X_s\prime$, which can be regarded as a linear part and a nonlinear part. Suppose $W_{tb}^T = (W^T, W_s^T)$, where W and W_s are the corresponding weights for vector X and $X_s\prime$ respectively. Therefore, we have

$$y_l = W^T X \tag{5.14}$$

$$y_s = W_s^T X_s\prime \tag{5.15}$$

$$y_{tb} = y_l + y_s \tag{5.16}$$

We can compare the contribution of y_l and y_s to the final output y_{tb}. If the contribution of y_s is bigger than y_l, the trained bilateral filter is regarded as working nonlinearly.

For the neural filter, the hyperbolic tangent function used in the hidden unit can be considered as an identity function in the input range of $[-0.1, 0.1]$ as shown in Fig. 5.11. If the input range is out of the linear range, then the neural filter works nonlinearly.
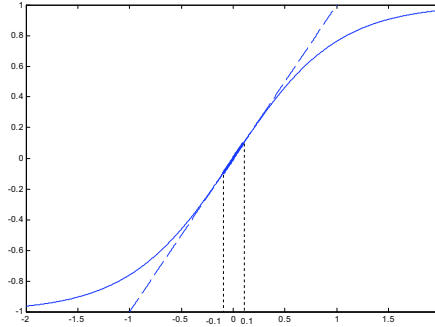

Figure 5.11: The input range of the hyperbolic tangent function. Solid line indicates the hyperbolic tangent function, dash line indicates the identity function and thick solid line shows the near-linear area of the hyperbolic tangent function.

In order to illustrate the nonlinearity, we perform the analysis on the compressed Motor image in the application of image de-blocking. Fig. 5.12 shows outputs of the nonlinearity analysis. The black area in the output indicates where the nonlinearity has been used and the white area indicates the opposite. As shown

in the figure, nonlinearity mainly occurs around edges in the image, where this nonlinearity has shown better edge preservation in the results.
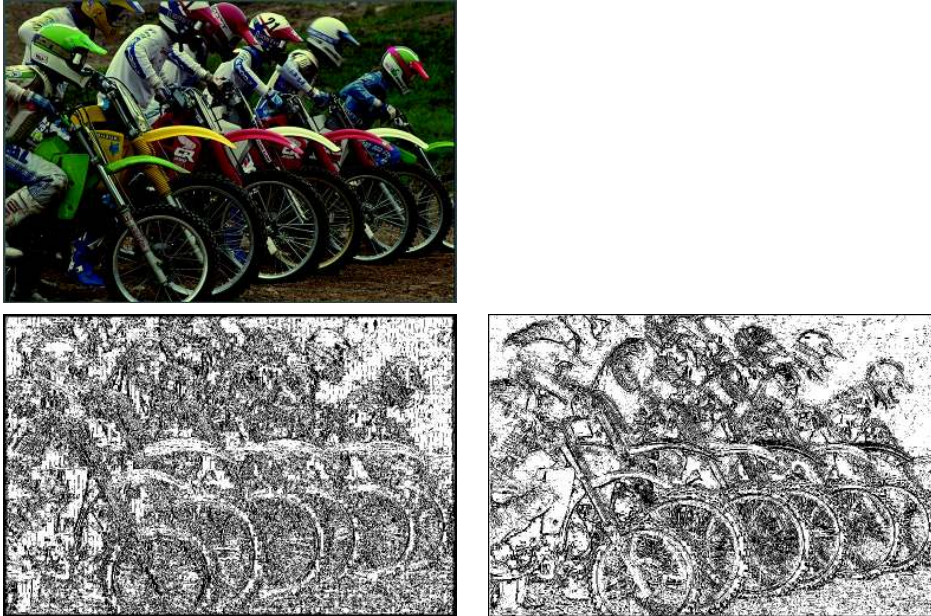

Figure 5.12: The output of the nonlinearity analysis, Top left: the input Motor image. Bottom left: the output of nonlinearity analysis from the trained bilateral filter. Bottom right: the output of nonlinearity analysis from the neural filter. The black area in the output indicates where the nonlinearity has been used.

5.5   Conclusion

In this chapter we have introduced several types of nonlinear filters. Different from the linear filter which only exploits the spatial information, the hybrid filter is proposed to incorporate the rank information and the neural filter uses a nonlinear transfer function to introduce nonlinearity. Inspired by the bilateral filter and the hybrid filter, we propose a new type of nonlinear filter, the trained bilateral filter. It utilizes the pixel similarity and spatial information as the original bilateral filter and can be optimized to acquire desired effects by the least mean square algorithm. These nonlinear filters are applied in the proposed framework of content adaptive filtering to see whether they can profit from the content classification. A thorough evaluation of these nonlinear filters is done in application to image de-blocking, noise reduction and image interpolation.

The chapter shows that given a filtering application like blocking artifact reduction it is possible to tune the bilateral filter to the optimal adaptation using the proposed trained bilateral filter. By adopting the similarity information, the trained bilateral filter possesses the essential characteristics of the original bilateral filter and can be optimized by the least mean square algorithm. It achieved the best results in the experiments. The order statistics filter can heavily suppress the noise but also destroys the details. The rank information only gives some indication about the pixel similarity, therefore the hybrid filter which combines the rank order and spatial information can profit little from it.

The neural filter has more flexibility and demonstrates the best performance to reconstruct the details. In the application of image interpolation, the rank order and similarity information do not bring much performance improvement. None of these filters is designed to inherently adapt itself to better image interpolation. Therefore, all the filters benefit a lot from the content classification, which is crucial for the interpolation.

All the nonlinear filters can profit from the content classification, but the trained bilateral filter profits little and it shows satisfactory edge preserving and noise reduction without the classification. This suggests that a nonlinear filter which inherently adapts to the signal well can perform better than a simple linear filter with the content classification. Furthermore, the results from the chapter suggest that designing a filter to inherently adapt to the signal is as important as designing content classifications. A good filter design can lead to a simpler content classifier. Taking account of the number of coefficients and the performance, the trained bilateral will be the best choice for the implementation.

The trained bilateral filter can adapt to noise or coding artifacts within the filter aperture. However, it can not adapt to the level of noise or coding artifacts because within the range of the filter aperture, the level of noise or coding artifacts is difficult to estimate. Using the histogram of local feature statistics to estimate the local image quality could improve the performance of the trained bilateral filter. Incorporating such adaption to the trained bilateral filter remains an interesting topic for future research.

Chapter 6

Trained Transfer Curves

In the previous chapters, we have seen how the proposed content adaptive processing framework can be used for different video enhancement applications such as noise reduction and resolution enhancement. And they have shown superior performance over other approaches. Until now, we have not yet discussed contrast enhancement. In this chapter, we try to find if contrast enhancement can benefit from the proposed framework of content adaptive video enhancement.

From the literature it is known that the grey level transformation is widely used for contrast enhancement. The transfer curve for the transformation can be selected from pre-defined functions or obtained by processing the histogram, such as histogram equalization. However, we did not find examples where the transfer curve function can be tuned to achieve some desired effects by a supervised learning.

As image content varies between different regions in the video, it is also very desirable to have *localized* contrast enhancement. However, how to optimize the local contrast enhancement is still a question.

To answer these questions, we propose a trained approach to obtain the optimal transfer curve for contrast enhancement, which is based on histogram classification. A supervised learning is applied to optimize the transfer curve from a version enhanced by computationally intensive algorithms. Furthermore, we propose a combined global and local contrast enhancement approach using separately trained transfer curves. A global transfer curve and a local one, are used to transform the local mean and the difference between the local mean and the processed pixel, respectively. The advantage is that it can adapt to both global and local content and offer optimized enhancement.

The rest of the chapter is organized as follows. We start with a brief survey of different contrast enhancement techniques in Section 1. Then we propose the trained transfer curve approach for the global enhancement in Section 2. Section 3 discusses content classification for the local enhancement. We present a hybrid

enhancement approach based on the trained global and local methods in Section 4. In order to evaluate the proposed method, subjective evaluation experiments have been performed and the results are presented. Finally, we draw our conclusion in Section 5.

6.1   Introduction

Contrast enhancement is an important image processing technique to increase the image quality. It was traditionally approached by the grey level transformation, which is one of the simplest of all image enhancement techniques [115]. It is a transformation that maps a pixel value in an input image to a pixel value in the processed image. Usually the values of the transformation are stored in one-dimensional array and the mappings are implemented by look-up-tables [113]. Early grey level transformations use some basic type of functions for image enhancement, such as linear and logarithmic functions [114].

Since a fixed transformation may not be optimal for different image contents, some histogram-based approaches have been proposed [117]. A histogram shows the frequency distribution of all the grey levels in an image, that is, the numbers of pixels with every grey level. In these approaches, the grey level transformation values are calculated through processing the histogram. One typical example is histogram equalization, which re-maps the grey levels in the image such that the resultant histogram approximates that of a uniform distribution. Fig. 6.1 shows that the contrast of an input image is well enhanced by histogram equalization. The histogram of the processed image is more uniformly distributed. The problem of histogram equalization is that it can over-enhance the image contrast when the grey scale distribution is highly localized [121]. It is not clear how to tune these approaches for some desired enhancement.

In the category of histogram equalization methods, a fixed transfer curve is usually applied globally to an image. This is based on the assumption that the image quality is uniform over all areas. However, this assumption does not hold when distributions of grey levels change from one region to another. Therefore, another category consisting of local contrast enhancement [120] [118] [119] has been proposed to improve the local enhancement performance. The main idea in the local contrast enhancement is to find the transformation function for every pixel based on its neighborhood content. In [120] the histogram of grey levels in a window around each pixel is generated first. The cumulative distribution of grey levels, that is the cumulative sum over the histogram, is used to map the input pixel grey levels to the output. However, these methods usually have some disadvantages that they often generate halo artifacts between different regions. An example of such halo artifacts is shown in Fig. 6.2. The halo artifacts occur near
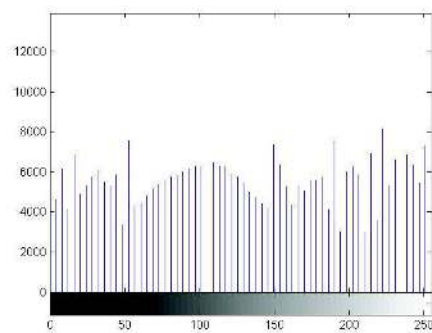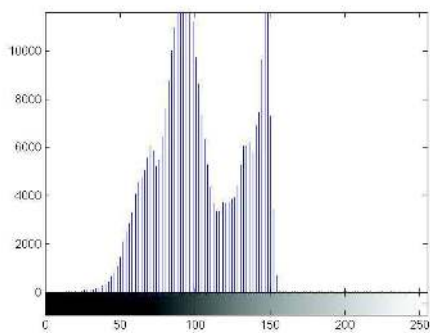
(A) Input   (B) Processed by histogram equalization

Figure 6.1: An example of histogram equalization: the image contrast is enhanced and the histogram of the processed image is more uniformly distributed.

the region boundaries since the local histograms there change dramatically.

The local contrast enhancement approaches do not have the knowledge of the global image content so that they can not enhance the overall contrast. Therefore, a third category includes hybrid methods which enhance the contrast at different resolution scales. One example is to use frequency band boosting. An image is split into different frequency bands and local contrast enhancement can be achieved by boosting the gains of the higher frequency bands. In [122] Bae proposed to separate an input image into a base part and a detail part through edge preserving filtering. The global contrast of the base part is enhanced by applying histogram matching from an example image. The details part is boosted and added back to the modified base. The halo artifacts are prevented, but the method lacks automatic content adaptation and it is also not clear how to optimize the enhancement.

6.2   Trained transfer curves for global enhancement

In order to tune the contrast enhancement to achieve some desired effects, such as manually tuned results by experts or superior enhancement by some computationally expensive algorithms, we propose a trained classification-based approach for contrast enhancement. Here we start with the global contrast enhancement, later introduce the local and hybrid enhancement. Existing global methods usually compute the transfer curve from the histogram on-the-fly and methods generating satisfactory results are often computationally expensive. We propose to obtain trained transfer curves based on histogram classification through a off-line training rather than computing them on-the-fly. In this approach, the histogram of an input image is first calculated and classified into a number of classes. Then in every class, we obtain the optimal transformation function using input images and desired target versions.

6.2.1   Proposed approach

To classify the histogram into a number of classes, we could try to use adaptive dynamic range coding (ADRC) mentioned in the previous chapter. ADRC is a simple and effective way to classify the image structure by thresholding the pixel vector. We could expect that similar thresholding could also lead to an effective histogram classification. Let $h(n)$ denote the histogram, where n is the bin number. Then we have

$$ADRC(h(n)) = \begin{cases} 0, & if \ h(n) < h_{av} \\ 1, & otherwise \end{cases} \qquad (6.1)$$
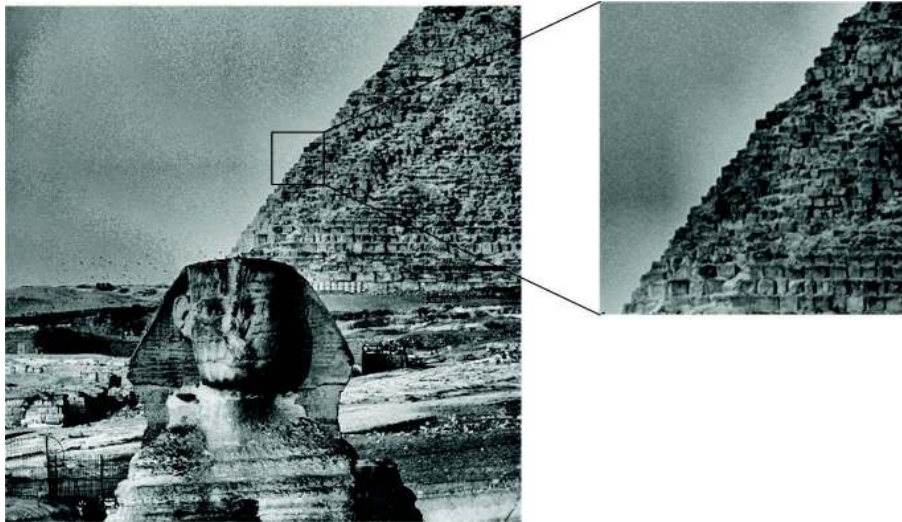
(A) Result from a local contrast enhancement method [118]
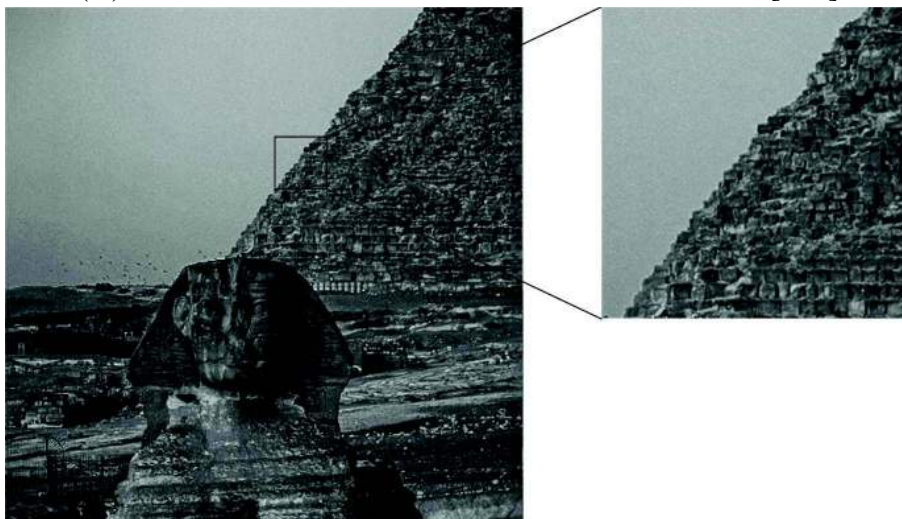
(B) Result from a hybrid contrast enhancement method [122]

Figure 6.2: Results from a local and hybrid contrast enhancement method. The local methods usually generate halo artifacts between different regions. This can be prevented by using edge preserving filtering to generate different resolution scales.
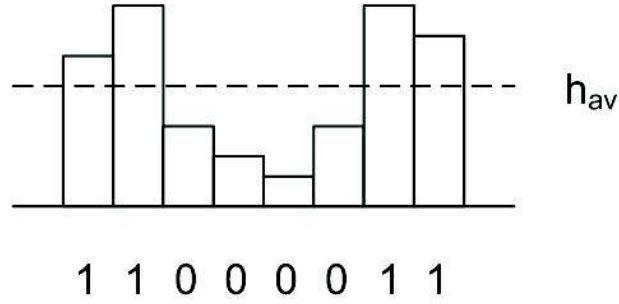
Figure 6.3: An example of the histogram classification.

where h_{av} is the average level in the histogram. To avoid an impractical number of classes, we use eight bins for the histogram calculation. The original luminance value is quantized equally into eight levels. The concatenation of the ADRC codes will give the class number, c. Fig. 6.3 shows an example of the histogram classification.

Let $X = \{X(i,j)\}$ denote a given image composed of L discrete grey levels denoted as $x_0, x_1, ..., x_{L-1}$, where $X(i,j)$ represents the intensity of the image at the spatial location (i,j) and $X(i,j) \in \{x_0, x_1, ..., x_{L-1}\}$. The class number for the histogram of the input image is c. Let $Y = Y(i,j)$ denote the desired reference image. Let $f_c(x)$ be the transfer curve function to be obtained for class c. Then we have the estimated output

$$\hat{X}(i,j) = f_c(X(i,j)) \tag{6.2}$$

The mean squared error MSE between the desired image and the estimated image will be

$$MSE = E[(Y - f_c(X))^2] \tag{6.3}$$

The optimal solution can be obtained by minimizing MSE. Taking the first derivative with respect to every variable in $f_c(x)$ and setting them to zeros, we obtain:

$$f_c(x) = E[Y(i,j)|c], \, i,j \in \{X(i,j) = x\} \tag{6.4}$$

The optimization procedure of the proposed trained transfer curve is shown in Fig. 6.4. We use images that need to be enhanced as the input images and apply the computationally intensive enhancement to generate reference images. These input images and output reference images are used as the training material. Before the training, the images are classified using the histogram classification. The image pairs that belong to one specific class are used for the corresponding training, resulting in an optimized transformation function for this class.
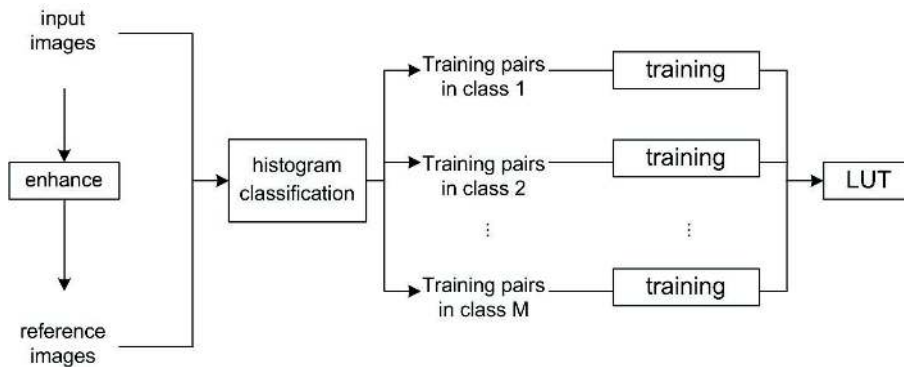
Figure 6.4: The training procedure to obtain the trained transfer curve. The input and output target image pairs are collected from the training material and are classified using the histogram classification. The transfer curves are then optimized for specific classes.
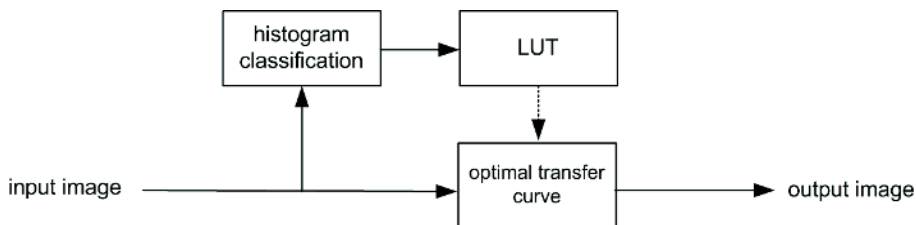

Figure 6.5: The block diagram of applying the classification-based transfer curve: the grey level histogram is classified using the ADRC and the optimal transfer curve for that class is obtained from the LUT.

To illustrate how to apply this classification-based transfer curve, a block diagram is shown in Fig. 6.5. First the histogram of the input image is obtained and classified by ADRC. Then the ADRC code is used to get the optimized transfer curve in that class from the look-up-table. The grey levels in the input image are transformed using the optimized transfer curve, resulting in the enhanced image.

6.2.2 Experimental results

For the evaluation of this proof-of-concept, we pick up the Auto Contrast function in Adobe Photoshop as the learning target, as it generates satisfactory results and can be regarded as an unknown method similar to manual tuning. In the experiment, an image database that contains various types of images has been used for the training. The desired target images are generated by applying the Auto Contrast function to the image database. For testing, we used the test images, Lena and Stone, as shown in the Fig. 6.6 and 6.7, which are not included in the training

Original Lena image
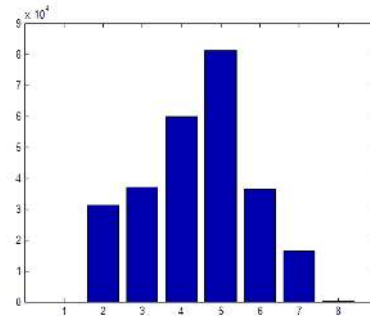
Histogram of Lena image

class code 01111100
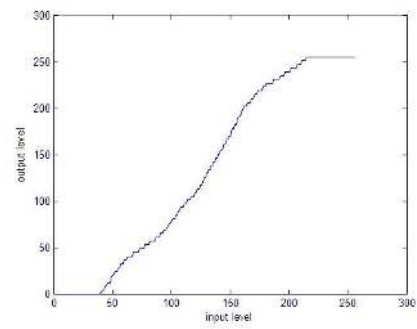
processed by Auto Contrast

Transfer curve defined by Auto Contrast

processed by the proposed method

Trained transfer curve

Figure 6.6: Experiment results on the test image Lena, its histogram classification, and enhanced results and transfer curves by Auto Contrast and the proposed method.

Original Stone image

Histogram of Stone image
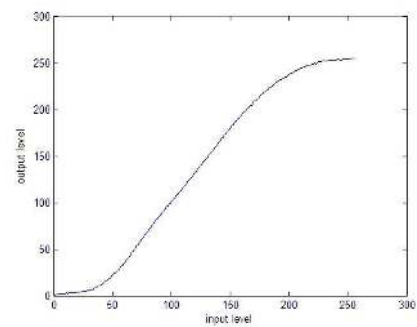
class code 01111100

processed by Auto Contrast

Transfer curve defined by Auto Contrast
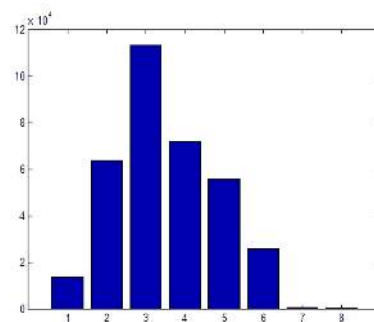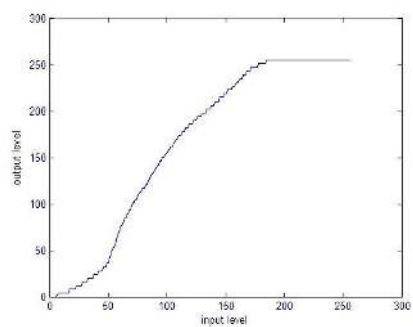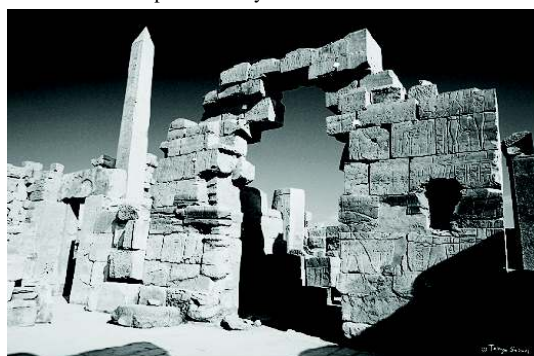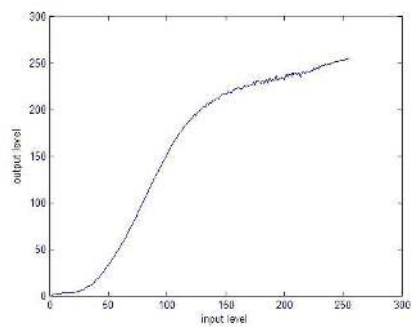

processed by the proposed method

Trained transfer curve

Figure 6.7: Experiment results on the test image Stone, its histogram classification, and enhanced results and transfer curves by Auto Contrast and the proposed method.
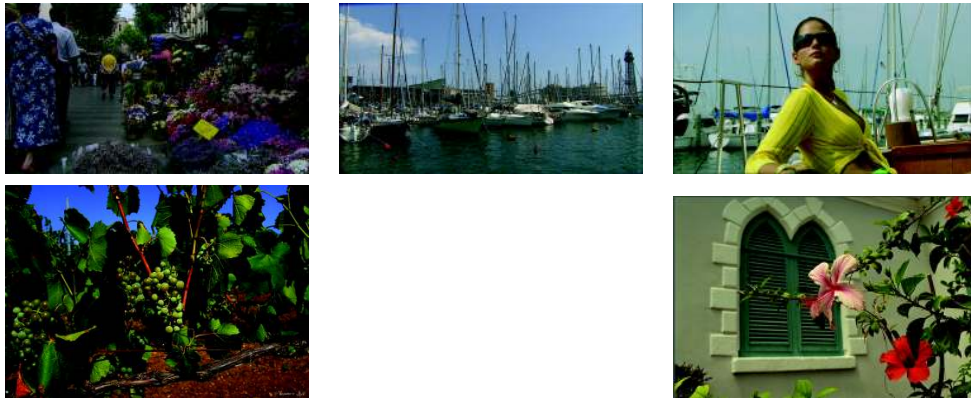
Figure 6.8: The testing material used for the subjective evaluation.

set.

As one can see, the results from the trained transfer curve method are quite close to the enhanced version by Photoshop Auto Contrast. More precisely, one can see that the trained transfer curve has a similar shape, but smoother. The result suggests that the desired effects of some enhancement methods can be mimicked by a supervised learning. As the histogram processing based approaches need to process the histogram, the proposed method does not require that. Therefore it has a lower complexity. We shall emphasize that the ADRC histogram classification is merely an example classification. Further research into alternative classification schemes is necessary. As a proof-of-concept, however, the ADRC classification is efficient and serves our needs.

In order to subjectively assess the proposed method, we performed a paired comparison of test sequences and their enhanced versions obtained from Auto Contrast and our trained approach. The test set includes two video sequences and three still images, both of which were not included in the training. The screen shots of these sequences and images are shown in Fig. 6.8. Taking the original material and its two enhanced versions as the test material, each two of them with the same content were shown side by side on an LCD screen with a resolution of 1920 by 1080 and the order had been randomized. Eighteen expert and non-expert viewers were asked to sit in front of the screen at a distance of three times the screen height and select the one that he/she perceived as having the better image quality.

An analysis of the paired comparison result as proposed by Montag [61] is shown in Fig. 6.9. The 95 percent confidence interval is used for the image quality scale. Here, we show the average results on all the sequences and images and also the results on the sequence group and the image group, respectively. On
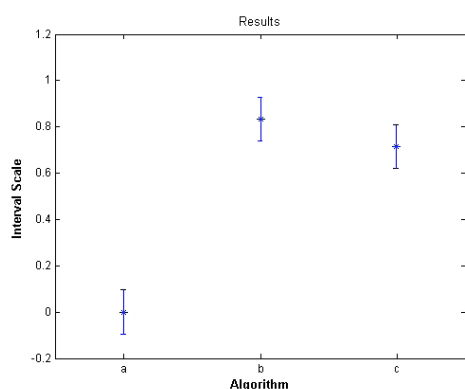
Results for all material
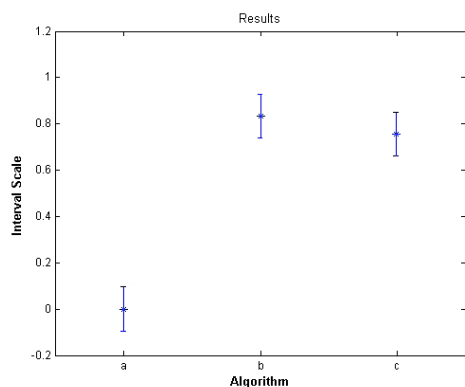
Results for sequences
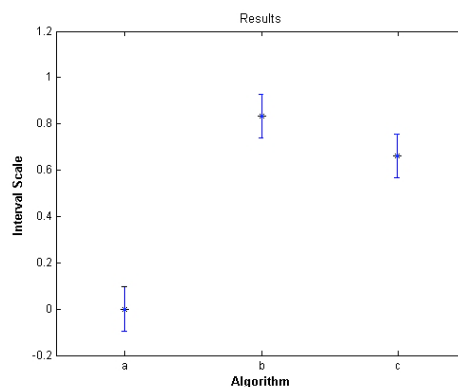
Results for images

Figure 6.9: The subjective evaluation results: A higher value on the quality scale means a higher preference by the viewers. Methods: a - original, b - the trained global approach, c - the trained global and local approach.

average, the quality scale of the two enhanced versions is higher than the original. This suggests that the perceptual image quality has been significantly increased by the two enhanced versions. Similar results have also been reflected in the sequence group and the image group. In the image group, the difference between the two enhanced versions is larger. Although there is no significant difference between the trained approach and the target algorithm, the trained approach is more preferred in the test. We expected this, because the trained approach uses the statistically averaged result and the trained transfer curve is smoother. However, more experiments are required to get statistically significant results.

6.3   Trained transfer curves for local enhancement

In the previous section, we have introduced the proposed transformation function to the global image enhancement, that is, a fixed transformation function is used to provide similar enhancement for all regions of the entire image. This is based on the assumption that the image quality is uniform over all areas. However, this assumption does not hold when distributions of grey levels change from one region to another. In such a case, a local contrast enhancement will improve the performance.

6.3.1   Local enhancement based on histogram classification

A straight-forward approach to the local enhancement is to use the mentioned global method for every pixel based on the histogram of its local neighbors. This has been proposed in [120]. For every pixel, the local histogram needs to be calculated and classified. However, this is computationally demanding. Therefore, it is less suitable for real-time video enhancement.

An alternative would be to apply the enhancement in a way that compromises between a global histogram equalization algorithm and a fully adaptive algorithm. In this case, the image can be divided into a limited number of non-overlapping blocks and the same histogram equalization technique is applied to pixels in each block. The problem of this approach is that the pixels near the block border will be mapped differently and consequently a significant blocking effect will be resulted. To alleviate the blocking effect, bilinear interpolation of the mappings of neighboring blocks can be used, that is, the mapping for a pixel is obtained by using a weighted-sum of the mappings of its four nearest blocks. Fig. 6.10 shows the results of the block based local histogram equalization with and without the bilinear interpolation. The results show that the local contrast can be enhanced, but the blocking effect still remains even after the bilinear interpolation. Therefore, we have to look for classifiers that could avoid using a block-based approach.

6.3.2   Local enhancement based on local mean and contrast

We consider that local mean and contrast in a neighborhood of the processed pixel could be a better choice than the local histogram for content classification. The local mean is defined as the local average of the neighboring pixels and the local contrast is defined as the difference between the maximum and minimum pixel values. As they can be computed through a sliding window, we could expect that it will reduce the blocking effect.

Therefore, we train the transfer curve based on the local mean and the local contrast. In the experiment the local mean M and the local contrast DR are

(A) Without interpolation   (B) With interpolation
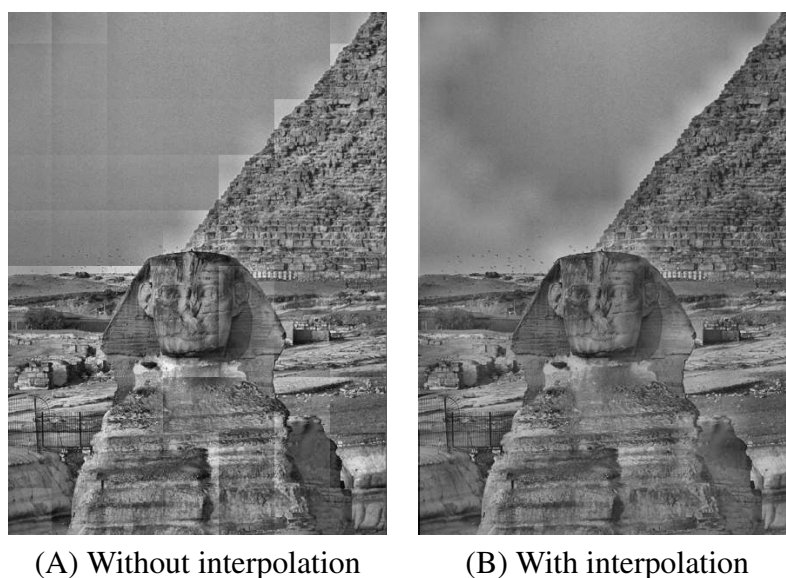
Figure 6.10: The block based local histogram equalization: (A) without interpolation, (B) with interpolation.

equally quantized into 8 levels, that is, 3 bits class code, respectively. The obtained transfer curve is shown in Fig. 6.11. The upper figure shows trained transfer curves for different local mean values with a fixed local contrast value. And the lower figure shows trained transfer curves for different local contrast values with a fixed local mean value.

The results suggest that the contrast enhancement does not depend much on the local mean, which means for the local contrast enhancement we could use only the local contrast classification. From the transfer curves for different local contrast with a fixed local mean, we can see that when the local contrast is higher, there will be more stretch in the local details. This suggests that in the smooth area the local details will not be stretched heavily to avoid some artifacts and in the fine structure area it will be boosted.

Different from the histogram classification, the local enhancement based on the local contrast is approached by a sliding window manner. Furthermore, it shows that local contrast enhancement does not depend on the local mean, through which we can integrate the global enhancement.
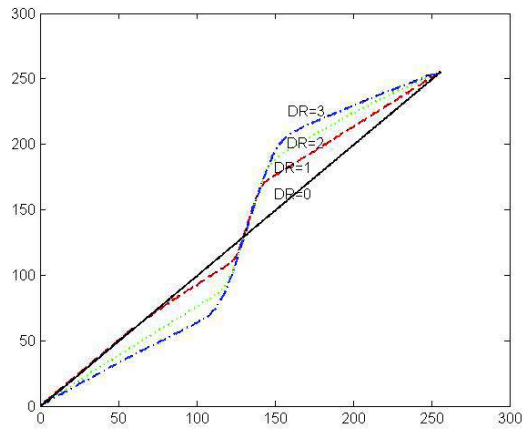
(A)
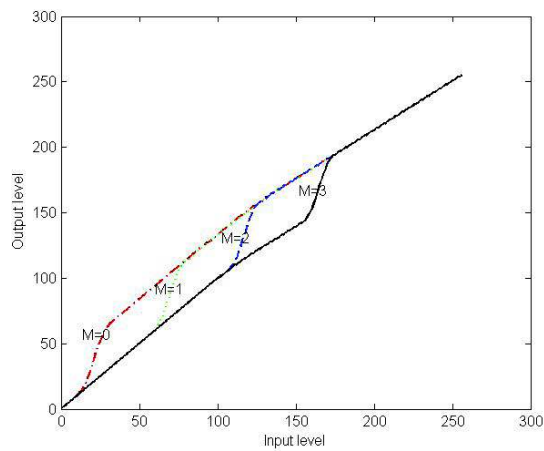
(B)

Figure 6.11: Trained transfer curves using the local mean and contrast classification: (A) Curves for different local contrast values with a fixed local mean value, (B) Curves for different local mean values with a fixed local contrast value.
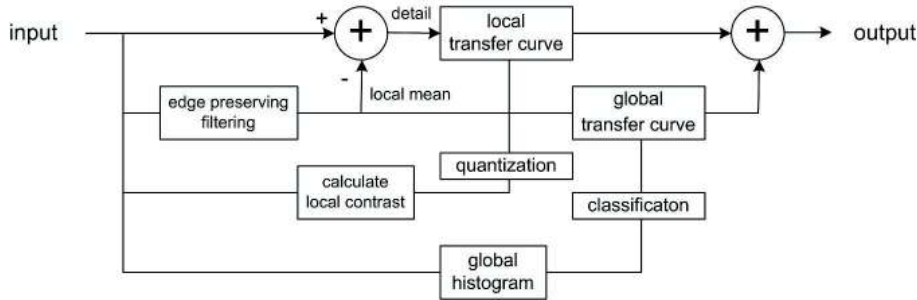
Figure 6.12: Block diagram of the proposed hybrid enhancement method: the input is divided into a local mean part and a detail part by using the edge-preserving filtering. The local mean part is transformed using a trained global curve based on the histogram classification, and the detail part is transformed by a trained local curve based on the local contrast quantization.

6.4 Trained transfer curves for hybrid enhancement

Since the global enhancement alone cannot enhance local details in an image and the local enhancement does not improve the overall contrast, a hybrid approach could be beneficial. Here, we propose a new contrast enhancement approach. Inspired by Bae's method [122], we divide the input image into two parts, a local mean part and a detail part. To improve the global contrast, the proposed method applies a trained global transfer curve to the local mean, which is obtained through edge-preserving filtering over the input image. The detail part, which is the difference between the local mean part and the original input, is transformed by a local transfer curve controlled by the local contrast.

6.4.1 Proposed approach

The diagram of the proposed method is shown in Fig. 6.12. First within a local window from the input image, the local mean is calculated using edge preserving filtering. The difference between the central pixel and the local mean is used as the input to the local transfer curve, which is selected according to the local contrast. The final output pixel is obtained by adding the transformed difference and local mean. The window slides pixel by pixel over the entire image and the processed image then is obtained.

The local mean $u(i, j)$ at position (i, j) is calculated within a window centered at the processed pixel. Let $X(i + m, j + n)$ denote the pixels within the window, where m, n are the horizontal and vertical offset to the pixel coordinates, respectively. To avoid generating halo artifacts, edge preserving filtering, such as the sigma filter and the bilateral filter, can be used to calculate the local mean. In the experiment we find that the sigma filter has similar performance as the bilateral

filter for this application and also the sigma filter has a lower complexity. There-fore, we use the sigma filter in the proposed approach. The sigma filter averages the pixels which are in the same region as the central pixel. Here, if the differ-ence between a pixel and the central pixel is below a certain threshold, then the pixel is regarded to be in the same region as the central pixel. Otherwise, it is not. Therefore, the local mean $u(i, j)$ is:

$$u(i, j) = \sum_{m,n} X(i + m, j + n)/N_k m, n \in \{|X(i + m, j + n) - X(i, j)| < th\}$$

where th is the mentioned threshold and N_k is the number of the pixels which belong to the same region as the central pixel. Then the local detail part $v(i, j)$, which is the difference between the central pixel and the local mean, is defined as:

$$v(i, j) = X(i, j) - u(i, j) \qquad (6.5)$$

The local detail $v(i, j)$ will be transformed using a local transfer curve, which is controlled by the local contrast. Let $LC(i, j)$ denote the local contrast, which is defined as the quantized difference between the maximum and minimum pixel value in the window with the same region label:

$$LC(i, j) = \frac{max(X(i + m, j + n)) - min(X(i + m, j + n))}{Q}$$
$$m, n \in \{|X(i + m, j + n) - X(i, j)| < th\} \qquad (6.6)$$

where Q is a pre-defined quantification.

The optimal local transfer curve can be obtained in a similar manner as the global transfer curve, except that the classification is based on the local contrast and the enhancement to generate the reference image is applied locally.

Let $g_{LC}(x)$ denote the local transfer curve and $f_{GC}(x)$ denote the global trans-fer curve, then the output pixel $Y(i, j)$ will be:

$$Y(i, j) = f_{GC}(u(i, j)) + g_{LC}(v(i, j)) \qquad (6.7)$$

6.4.2   Experimental results

For the evaluation, we test our proposed method on some natural images, and we also compare the global and local methods from the previous sections. For the proposed method, we obtained the local transfer curve by learning from the adap-tive contrast limited histogram equalization method [118] which is well-known and produces superior results. For the global histogram classification, we use a 6 bin histogram calculation, that is, 6 bits for the classification. For the local con-trast classification, we quantize the local contrast into 32 levels, which are 5 bits for the classification. Different window sizes have been used in the experiment.

We show some experimental results using Fig. 6.1 (A) as an input image. Image fragments from the input image and different processed versions are shown in Fig. 6.13. We can see that the global method can enhance the overall contrast well, but the local contrast enhancement is rather limited. The result of the local histogram equalization method can increase the local contrast very nicely in the texture area. However, it generates quite visible halo artifacts in the boundary area between the sky and the pyramid. When the block size of the local method increases, the halo artifact becomes less visible but also the local contrast boost becomes weaker. The hybrid method seems to be able to combine the advantages of the global and local methods. It can improve the overall contrast while boosting the local details. With a local window size of 5 pixels, it can only boost the details near the highest frequency. When the window size grows to 21 pixels, it can also boost lower frequency. Taking all of these into consideration, it seems that the hybrid method with a larger window size is the best approach.

As there is no method for a quantitative evaluation, we performed a subjective assessment on the hybrid method compared with the trained global method and the original, similar to the previous section. The same test material shown in Fig. 6.8 is used. As suggested, we use a window size of 21 by 21 pixels to obtain well balanced result. The subjective evaluation results are shown in Fig. 6.14. Here, we can see that, on average, the hybrid approach shows a significant improvement over the global enhancement, which is significantly better than the original input. Also it seems that the global enhancement is more appreciated in the experiment since the improvement of the global enhancement is greater than that of the local enhancement. This is probably because the local detail enhancement is less visible than the overall contrast enhancement.

6.5 Conclusion

In this chapter, we have shown a proof-of-concept for the trained approach to obtain contrast enhancement by a supervised learning. The transfer curve depends on the histogram classification of the input image. It shows that it is possible to obtain the desired effect by learning other histogram-based enhancement methods or expert-tuned examples using the trained approach. We have initially used the ADRC as a histogram classifier, which is shown by the experimental result to be *effective* for the enhancement. Whether it is the *optimal* histogram classification requires further research work.

For the local enhancement, we conclude that it does not depend on the local mean, but on the local contrast. Therefore, the coarsely quantized local contrast is proposed as the classifier. Furthermore, we have introduced a hybrid method based on the trained approach. The input image is divided into a local mean part
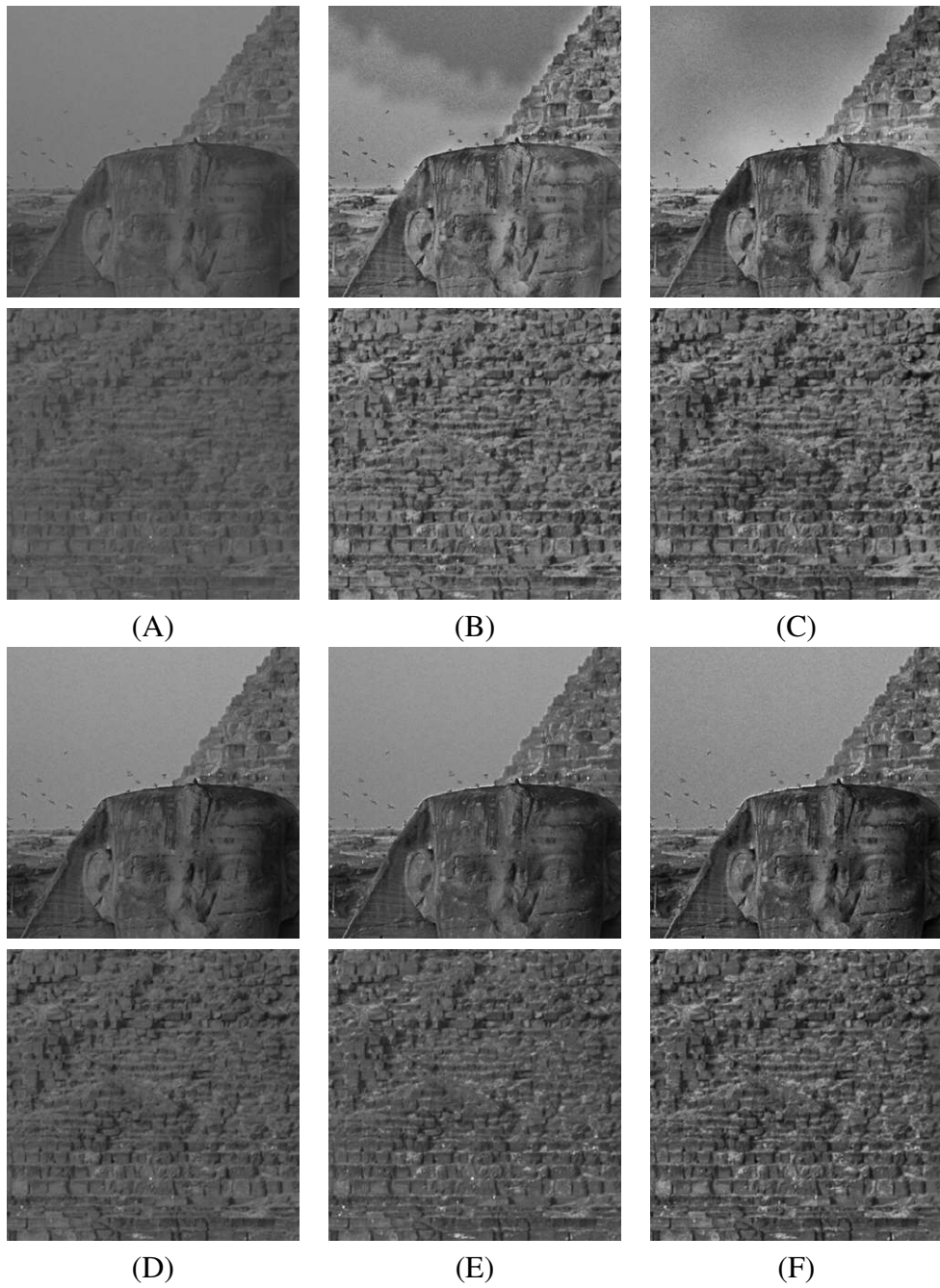
Figure 6.13: Image fragments from the processed results: (A) input, (B) enhanced by local method with a block size of 16 by 16 pixels, (C) enhanced by local method with a block size of 64 by 64 pixels, (D) enhanced by global method, (E) enhanced by hybrid method with a local window size of 5 by 5 pixels, (F) enhanced by hybrid method with a local window size of 21 by 21 pixels.
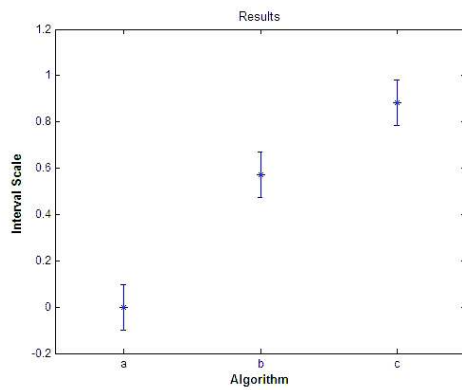
Results for all material
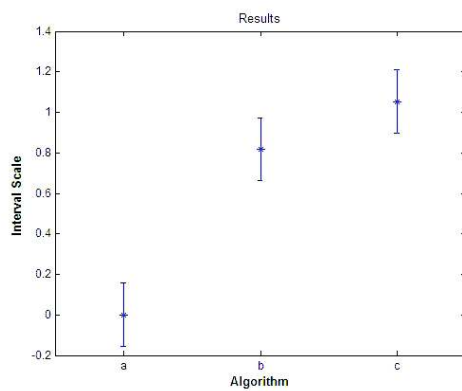
Results for sequences
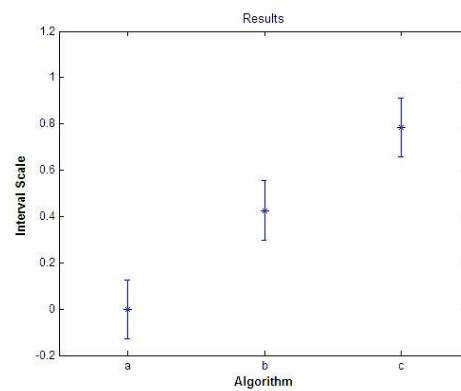
Results for images

Figure 6.14: The subjective evaluation results: A higher value on the quality scale means a higher preference by the viewers. Methods: a - original, b - the trained global approach, c - the trained global and local approach.

and a detail part by using the edge-preserving filtering to prevent the halo effect. The local mean part is transformed using a trained global curve based on the histogram classification, and the detail part is transformed by a trained local curve based on the local contrast classification. Compared with the global and local methods, the hybrid method shows the best balanced performance at enhancing the image contrast and preventing halos. The proposed hybrid approach seems to be promising. However, in this chapter only a two-scale version is investigated. We could expect that an extension to more scales will further improve the enhancement performance, which merits further attention.

Chapter 7

Conclusions and future work

There are various research topics in the field of video enhancement, such as noise reduction, sharpness, resolution, color and contrast enhancement. Although quite diverse in terms of interests, all these topics have one common ultimate goal, that is, to improve the subjective video quality, or make videos or images more beautiful. How to achieve it is not always trivial. In the past, algorithm designers have to iterate between the processes of evaluating the algorithm improvement through subjective assessment experiments and adjusting the algorithm improvement, which are very time consuming and eventually lead to a high design cost. Researchers try to find a subjective video quality metric to replace subjective assessment experiments. Such attempts have not been successful so far.

Least mean square methods that optimize parameters automatically through minimizing the mean square error between processed images and desired versions attract researchers' attention. It can replace the parameter tuning by heuristics. However, these methods often have a solution which is optimal on the average image quality. They perform rather poorly on local image details. In order to solve this problem, some classification based approaches have been proposed, which apply the least mean square optimization for individual categories that are classified by local image content, such as edge directions. Such local content adaptivity has led to a better image quality.

Having this concept of local content adaptivity for least mean square optimization, we find that it could be generalized into an ideal framework for content adaptive video processing. The goal of this thesis is to provide such a content adaptive video processing framework, which can replace heuristics and reduce the algorithm design time. We recognize two main parts in the concept: video content classification and processing model selection. To generalize the framework, we have proposed more new classifiers and new types of processing models in these two main parts.

Prior work about this concept mainly focused on resolution enhancement such

as resolution up-scaling and de-interlacing. Not much research work has been done with noise reduction, especially coding artifact reduction in digitally coded videos. To extend the framework to cope with coding artifact robust enhancement applications, a classifier to detect coding artifacts is required. We find that the combination of local structure and contrast information can provide a simple but effective means to distinguish coding artifacts from real image structures. Incorporating this new classifier into the content adaptive filtering framework has led to several artifact robust enhancement algorithms for digitally coded videos, including artifact reduction and its integrations with sharpness and resolution enhancement.

Incorporating these new features in the framework not only improves the enhancement but also introduces more interesting applications to the framework. We further explore the content classification for blurred videos, which has resulted in new local blur estimation methods and two blur dependant enhancement applications. Additionally, we show that it is possible to reduce the number of content classes without much sacrifice to the performance.

Usually a linear model or filter is used for such a concept in the prior work. It is known that nonlinear filters can perform better where the nonlinearity is significant, such as in smoothing tasks where edge preservation is important. It is interesting to see what the nonlinear filters could add to the framework. We have extended the filter model to the hybrid filter, the neural filter. Our exploration in this area has also resulted in a new type of nonlinear filter, trained bilateral filter which possesses the advantages of the original bilateral filter and the least mean square filter. Besides filtering, there are some other processing forms, such as using transfer curves in the contrast enhancement. We have also explored the opportunity to apply the framework to contrast enhancement by using a content adaptive transfer curve in the adaptive processing part.

7.1 Conclusions

In this thesis, a versatile framework for content adaptive video processing is proposed. We have proposed new classifiers and processing models to the framework and extend it for more applications. The main findings are summarized as follows.

7.1.1 Content classification

Chapter 2 presents a novel classifier for coding artifacts. Based on this classifier, several coding artifacts reduction algorithms combined with resolution up-conversion and sharpness enhancement have been developed.

- Using two orthogonal image properties, local structure and local contrast, can distinguish the coding artifacts. Usually many existing coding artifact reduction methods require the compression parameter or the bit stream information or coding block grid detection to obtain good performance. The proposed classifier does not require such information, which makes it suitable for postprocessing digital video compressed by different codec standards.

- In the application of JPEG de-blocking, the proposed coding artifact reduction algorithm showed better performance than Nosranatia's method which is one of the best methods. In the application of H.264 de-blocking, the proposed method outperforms the in-loop filter recommended in the H.264 standard, although the in-loop filter has an advantage to utilize all the compression information in the loop.

- We have proposed an integrated artifacts reduction and resolution up-conversion approach. Based on the proposed coding artifact classification, the proposed framework is used for estimating the high resolution pixels and eliminating the coding artifacts at the same time. The optimal coefficients are obtained by a training between the high resolution reference and the simulated degraded version. The results showed that the proposed method can well reduce coding artifacts and preserve image structures and fine details. We have also compared the proposed integrated approach with two alternative concatenated versions. The results showed that the integrated approach not only has a better performance but also is much more economical than the other two in terms of coefficients look-up-table size. Similarly, we have also proposed an integrated approach to achieve simultaneous coding artifact reduction and sharpness enhancement.

In Chapter 3, we look at the problem of enhancing videos which contain differently blurred objects. New local blur estimation methods have been proposed. Furthermore we have applied the proposed blur estimation to two blur dependant video enhancement applications.

- The maximum of the difference ratio between an original image and its two re-blurred versions has been shown to be able to identify the edges and estimate the local blur radius in the original image. Compared to other edge based approaches, it does not require edge detection and have robust estimation, especially for the interference from neighboring edges.

- To obtain a consistent blur estimation over objects that allows more stable restoration or enhancement, we have proposed three approaches to improve

the consistence of the blur map, spatial-temporal neighborhood, propagating estimates and segmentation-based approach. The proposed segmentation-based approach gives the most consistent and accurate estimation results.

- In the proposed segmentation-based approach, we include not only the color information but also the spatial information and blur radius into the feature space, which leads to meaningful segmentation results. We also see that some over segmentation problems which later can be partially solved by the blur estimation. By assigning the segmentation blur, the segments with the same blur radius are merged together in one region.

- We have applied the proposed segmentation based blur estimation in two applications, focus restoration and blur dependent coding artifact reduction. In the focus restoration application, experiments shows that objects that are differently blurred can be brought back into focus with the proposed method. In the application of blur dependent coding artifact reduction, the proposed approach has shown a better ability to remove the blocking artifacts in the unfocused part and a better sharpness enhancement in the focused part.

Chapter 4 discusses how to reduce the class-count in the content classification, which can increase exponentially with added image features.

- We have proposed three class-count reduction techniques, class-occurrence frequency, coefficient similarity and error advantage, which are based on our proposed framework and are generally applicable for different content classifications.

- In the applications of coding artifact reduction and image interpolation, it has been shown that these three class-count reduction techniques can greatly reduce the number of content classes without sacrificing much performance and are promising for applications with a large number of classes. Among them, the coefficient similarity approach and the error advantage approach produce the best result. Taking the cost into consideration, the class-occurrence frequency approach seems to be the choice.

7.1.2 Processing model

Chapter 5 introduces several types of nonlinear filters for the content adaptive processing framework.

- We have replaced the linear filter in the framework with several types of nonlinear filters. An evaluation of these nonlinear filters in application of image de-blocking, noise reduction and image interpolation showed that the nonlinear filters have better performance than the linear filter in different applications. All the nonlinear filters can profit from the content classification accordingly.

- We have proposed a new type of nonlinear filter, the trained bilateral filter. It utilizes the pixel similarity and spatial information as the original bilateral filter and can be optimized to acquire desired effects by the least mean square optimization.

- The good performance of the trained bilateral filter in noise reduction suggested that a single nonlinear filter which inherently adapts the signal well can perform better than a simple linear filter with the content classification.

Chapter 6 presents a proof-of-concept for the trained approach to obtain desired contrast enhancement.

- We have proposed to obtain trained transfer curves based on histogram classification through a off-line training rather than computing them on-the-fly. In this approach, the histogram of an input image is first calculated and classified into a number of classes. Then in every class, we obtain the optimal transformation function using input images and desired target versions. The result suggested that the desired effect of some enhancement methods can be mimicked by a supervised learning.

- We have also investigated the trained transfer curves for local contrast enhancement. The results suggested that the contrast enhancement does not depend much on the local mean, which means for the local contrast enhancement we could only use the local contrast classification.

- We further propose a hybrid method to enhance both the local and global contrast. The input image is divided into two parts, a local mean part and a details part. To improve the global contrast, the proposed method applies a trained global transfer curve to the local mean, which is obtained through edge-preserving filtering over the input image. The details part, which is the difference between the original input and the local mean part, is transformed by a trained local transfer curve controlled by the local contrast. Experiments showed that the hybrid method seems to be able to combine the advantages of the global and local methods. It can improve the overall contrast while boosting the local details.

7.1.3   Concluding remarks about the framework

To achieve content adaptivity, many image restoration techniques use prior information, or constraints about the local content in their cost functions. Usually these constraint functions have different optimization methods, which makes it difficult to find a unified approach. The proposed framework resembles the concept of prior information, or constraints, by quantizing these types of information into a number of content classes, which offers a unified way to incorporate local content information. By simply adding more classes, the framework can easily cope with, even integrate, more different applications, which greatly reduces the time to design a video enhancement algorithm.

To obtain the optimal solution, iterative approaches are usually taken to solve many classic inverse problems in the video processing field. The proposed framework uses a direct solution which can be regarded as a quantized approximation. The approximation can be linear or nonlinear, filters or transfer curves. The approximation is pre-calculated through an off-line training by minimizing the difference between output images and desired versions. No on-the-fly optimization is required. Therefore, it is more attractive for real-time implementation.

We also see content classification and adaptive processing as the two essential parts that supplement each other. Designing a processing model to inherently adapt to the signal is as important as designing content classifications. A dedicated model design can lead to simpler content classification. Vice versa, a good content classification can simplify the model design.

7.2   Future work

7.2.1   Introduce the framework to more applications

We have proposed new content classifiers and processing models to bring the framework to many applications, including coding artifact and noise reduction, sharpness, resolution and contrast enhancement. There are also some applications we have not tried with the proposed framework, for instance, color enhancement. We would expect that new classifiers need to be designed for these applications.

In Chapter 5, we have investigated nonlinear filters in the application of Gaussian noise reduction. Impulsive noise, which typically due to dusts inside the camera, or faulty CCD elements within digital cameras, is another different type of noise. It is also interesting to investigate further whether these nonlinear filters can bring similar benefits for reducing the impulsive noise.

7.2.2 Replace heuristics in classification

Research in the field has been greatly accelerated by the generic framework and automatic optimization of parameters. However, on the road towards a fully automatic video enhancement framework, there are problems that are not completely solved yet. Heuristics remains in the classification. Researchers still need to spend time to invent good classifiers. How to achieve automatic content classification for different application is still not known. In Chapter 3, we have proposed to cluster similar classes into larger classes. One solution could be using this clustering technique to form classes automatically from a large collection of basic classes. There has been also research work about using adaptive boosting [124] to synthesizing weak classifiers. It has been used to build a local coding artifact metric [125]. Such techniques are interesting for the future research for the content adaptive framework.

Bibliography

[1] P. A. Keller, *The cathode-ray tube: technology, history and applications*, Palisades Press, 1991, ISBN 0963155903.

[2] J. A. Castellano, *Handbook of display technology* Academic, 1992, ISBN 0121634205.

[3] J. Hutchison, "Plasma display panels: the colorful history of an Illinois technology", *ECE alumni news, university of Illinois*, vol. 36(1), 2002.

[4] H. Uchiike and T. Hirakawa, "Color plasma displays", *Proceedings of the IEEE*, vol. 90, no. 4, pp. 533-539, Apr. 2002.

[5] A. M. Tekalp, *Digital video processing*, Prentice Hall, 1995, ISBN 0131900757.

[6] A. Mohammed, C. N. Canagarajah and D. R. Bull, *Video coding for mobile communications*, Academic Press, 2002, ISBN 0120530791.

[7] C. Poynton, *Digital video and HDTV algorithms and interfaces*, Morgan Kaufmann, 2003, ISBN 1558607927.

[8] S. Firestone, T. Ramalingam and S. Fry, *Voice and video conferencing fundamentals*, Cisco press, 2007, ISBN 1587052687.

[9] S. Kawamura, "Capturing images with digital still cameras", *IEEE Micro*, vol. 18, no. 6, pp. 14-19, Dec. 1998.

[10] W. C. O'Mara, *Liquid crystal flat panel display: manufacturing science and technology*, Van Nostrand Reinhold, 1993, ISBN 0442014287.

[11] H. Kawamoto, "The history of liquid-crystal displays", *Proceedings of IEEE*, vol. 90, no. 4, pp. 460-500, Apr. 2002.

[12] S. Matsumoto, *Electronic display devices*, John Wiley & Sons, Mar. 1990, ISBN 0471922186.

[13] L. MacDonald and A. C. Lowe, *Display systems: design and applications*, John Wiley & Sons, Apr. 1997, ISBN 0471958703.

[14] D. E. Mentley, "State of flat-panel display technology and future trends", *Proceedings of the IEEE*, vol. 90, no. 4, pp. 453-459, Apr. 2002.

[15] H. Uchiike and T. Hirakawa, "Historical view and current status of plasma displays", *Industry Applications Conference*, vol. 1, pp. 45-51, Oct. 2003.

[16] J. Lee, D. N. Liu and S. Wu, *Introduction to flat panel displays*, John Wiley & Sons, Nov. 2008, ISBN 0470516935.

[17] T. Shigeta, "Improvement of moving-video image quality on PDPs by reducing the dynamic false contour", *SID Digest*, vol. 29, pp. 287-299, 1998.

[18] W. Song, X. Li and Y. Zhang, "Motion-blur characterization on liquid-crystal displays", *Journal of SID*, vol. 16, pp. 587-597, 2008.

[19] M. Klompenhouwer, *Flat panel display signal processing: analysis and algorithms for improved static and dynamic resolution*, Eindhoven University Press, 2006, ISBN 9074445756.

[20] O. Prache, "Active matrix molecular OLED microdisplays", *Displays*, vol. 22, no. 2, pp. 49-56, May 2001.

[21] R. Fink, "A closer look at SED, FED technologies", *EE Tines-Asia*, pp. 1-4, Aug. 2007.

[22] G. Crawford, *Flexible flat panel displays*, John Wiley & Sons, Apr. 2005, ISBN 0470870488.

[23] G. Moore, "Cramming more components onto integrated circuit", *Electronics Magazine*, vol. 38, no.8, Aug. 1965.

[24] "Moore's law", From Wikipedia, the free encyclopedia, available at: http://en.wikipedia.org/wiki/Moore's_law.

[25] J. L. Hennessy, D. A. Patterson and D. Goldberg, *Computer architecture: a quantitative approach*, Third Edition, Morgan Kaufmann, May 2002, ISBN 1558605967.

[26] S. Kilts, *Advanced FPGA design: architecture, implementation, and optimization*, Wiley-IEEE press, 2007, ISBN 0470054379.

[27] A. Jerraya, H. Tenhunen, and W. Wolf, " Multiprocessor systems-on-chips", *IEEE Computer Magazine*, vol. 38, no. 7, pp. 36-40, Jul. 2005.

[28] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview", *IEEE Signal Processing Magazine*, vol. 20, pp. 21-36, May 2003.

[29] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution", *International Journal of Imaging Systems and Technology*, vol. 14, no. 2, pp. 47-57, Aug. 2004.

[30] P. Debevec and M. Jitendra, "Recovering high dynamic range radiance maps from photographs", *SIGGRAPH Proceedings*, pp. 130-135, May 1997.

[31] D. J. Heeger and J. R. Bergen, "Pyramid based texture analysis/synthesis", *Computer Graphics Proceedings*, p. 229-238, 1995.

[32] W. T. Freeman, T. R. Jones and E. C. Pasztor, "Example-based super-resolution", *IEEE Computer Graphics and Applications*, vol. 22, pp. 56-65, May 2002.

[33] L. Estevez, "Real-time video surveillance system architecture", *Proceedings of SPIE Electronic Imaging Conference on Real-Time Imaging*, vol. 4303, pp. 19-26, Apr. 2001.

[34] H. Broers, W. Caarls, P. Jonker, and R. Kleihorst, "Architecture study for smart cameras", *Proceedings of the European Optical Society Conference on Industrial Imaging and Machine Vision*, pp. 39-49, Jun. 2005.

[35] M. Gamadia, V. Peddigari, N. Kehtarnavaz, G. Cook, and S. Lee, "Real-time implementation of autofocus on the TI DSC processor", *Proceedings of SPIE Electronic Imaging Conference on Real-Time Imaging*, vol. 5297, pp. 10-18, May 2004.

[36] N. Kehtarnavaz, H. Oh, and Y. Yoo, "Development and real-time implementation of auto white balancing scoring algorithm", *Journal of Real-Time Imaging*, vol. 8, no. 5, pp. 379-386, Oct. 2002

[37] I. Ozer, T. Lu, and W. Wolf, "Design of a real-time gesture recognition system", *IEEE Signal Processing Magazine*, vol. 22, no. 3, pp. 57-64, May 2005.

[38] X. Desurmont, B. Lienard, J. Meessen, and J. Delaigle, "Real-time optimizations for integrated smart network camera", *Proceedings of SPIE Electronic Imaging Conference on Real-Time Imaging*, vol. 5671, pp. 85-92, Feb. 2005.

[39] Q. Ji and X. Yang, "Real-time eye, gaze, and face pose tracking for monitoring driver vigilance", *Journal of Real-Time Imaging*, vol. 8, no. 5, pp. 357-377, Oct. 2002.

[40] H. Rowley, S. Baluja and T. Kanade, "Neural network- based face detection", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 203-208, 1996.

[41] A. Batur, B. Flinchbaugh, and M. Hayes III, "A DSP-based approach for the implementation of face recognition algorithms", *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 253-256, Apr. 2003.

[42] L. Chen and C. Grecos, "Fast skin color detector for face extraction", *Proceedings of SPIE Electronic Imaging Conference on Real-Time Imaging*, vol. 5671, pp. 93-101, Feb. 2005.

[43] H. Luong, B. Goossens and W. Philips, "Image upscaling using global multimodal priors", *Proceedings of the 9th International Conference on Advanced Concepts for Intelligent Vision Systems*, Volume LNCS 4678 , pp. 461-472, Springer, Aug. 2007.

[44] G. de Haan, *Video processing for multimedia systems*, Second Edition, Eindhoven University Press, 2000, ISBN 9090140158.

[45] R. J. Sypula, "Video noise reduction circuit", US Patent 4009334, 1977

[46] R. G. Keys, "Cubic convolution interpolation for digital image processing", *IEEE Transactions on Acoustics Speech, Signal Processing*, vol. ASSP-29, no. 6, pp. 1153-1160, Dec. 1981.

[47] M. Zhao and G. de Haan, "Intra-field de-interlacing with advanced upscaling methods", *Proceedings of the 8th International Symposium on Consumer Electronics*, pp. 315-319, Sep. 2004.

[48] M. Zhao and G. de Haan, "Content adaptive vertical temporal filtering for de-interlacing", *Proceedings of the 9th International Symposium on Consumer Electronics 2005*, pp. 69-73, Jun. 2005.

[49] A. K. Katsaggelos, J. Biemond, R. W. Schafer and R. M. Mersereau, "A regularized iterative image restoration algorithm", *IEEE Transactions on Signal Processing*, vol. 39, no. 4, pp. 914-929, Apr. 1991.

[50] D. C. Youla and H. Webb, "Image restoration by the method of convex projections", *IEEE Transactions on Medical Imaging*, vol. 1, no. 2, pp. 81-94, Oct. 1982.

[51] T. Jarske, P. Haavisto, and I. Defe, "Post-filtering methods for reducing blocking effects from coded images", *IEEE Transactions on Consumer Electronics*, pp. 521-526, Aug. 1984.

[52] H. C. Reeve and J. S. Lim, "Reduction of blocking artifacts in image coding", *Optical Engineering*, vol. 23, pp. 34-37, Jan./Feb. 1984.

[53] W. B. Pennebaker and J. L. Mitchell, *JPEG still image data compression*, Van Nostrand Reinhold, 1993, New York, ISBN 0442012721.

[54] M. Yuen and H. R. Wu, "A survey of hybrid MC/DPCM/DCT video coding distortions", *Signal Processing*, vol. 70, pp. 247-278, 1998.

[55] S. Minami and A. Zakhor, "An optimization approach for removing blocking effects in transform coding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, pp. 74-82, Apr. 1995.

[56] G. Lakhani and N. Zhong, "Derivation of prediction equations for blocking effect reduction", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 415-418, Apr. 1999.

[57] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color image", *Proceedings of IEEE International Conference on Computer Vision*, Bombay, India, Sept. 1998.

[58] F. Alter, S. Y. Durand and J. Froment, "Deblocking DCT-based compressed images with weighted total variation", *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 221-224, May 2004.

[59] S. Wan, M. Mrak and E. Izquierdo, "Perceptually adaptive joint deringing deblocking filtering for scalable video coding", *2nd International Mobile Multimedia Communications Conference*, Alghero, Sardinia, Italy, Sept. 2006.

[60] J. O. Limb and C. B. Rubinstein, "On the design of quantizer for DPCM coders : A functional relationship between visibility, probability and masking", *IEEE Transaction on Communication*, vol. 26, pp. 573-578, May 1978.

[61] E. D. Montag, "Louis Leon Thurstone in Monte Carlo: creating error bars for the method of paired comparison", *Proceeding of SPIE*, vol. 5294, pp. 222-230, Jan. 2004.

[62] ITU-T Recommendation and International Standard of Joint Video Specification (ITU-T Rec. H.264/ISO/IEC 14 496-10: 2005), Dec. 2005.

[63] A. Nosratinia, "Denoising of JPEG images by re-application of JPEG", *Journal of VLSI Signal Processing*, vol. 27, no. 1, pp. 69-79, 2001.

[64] A. Zakhor, "Iterative procedures for reduction of blocking effects in transform image coding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 2, no. 1, pp. 91-95, Mar. 1992.

[65] H. Paek, R. Kim, and S. Lee, "On the POCS-based postprocessing technique to reduce the blocking artifacts in transform coded images", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.8, no. 3, pp.358-367, Jun. 1998.

[66] Y. Yang, N. Galatsanos, and A. Katsaggelos, "Regularized reconstruction to reduce blocking artifacts of block discrete cosine transform compressed images", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, pp. 421-432, Dec. 1993.

[67] A. Pentland, "A new sense for depth of field", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 4, pp. 523-531, July 1987.

[68] A. Pentland, T. Darrell, M. Turk and W. Huang, "A simple real time range camera", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 256-261, 1989.

[69] M. Goncalves and F. Ernst, "Single-image motion and camera blur identification", Technical report, Philips Research, Eindhoven, 2005. PR-TN-2005/00298.

[70] Aron Beurskens, "Low cost robust blur estimator", Technical report, Philips Research, Eindhoven, 2007.

[71] G. de Haan, P. W. A. C Biezen, H. Huijgen, and O. A. Ojo, "True motion estimation with 3-d recursive search block-matching", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.3, pp. 368-388, Oct. 1993.

[72] R. L. Lagendijk, A. M. Tekalp, and J. Biemond, "Maximum likelihood image and blur identification: a unifying approach", *Optical Engineering*, vol. 28 (5), pp. 422-435, 1990.

[73] G. Pavlovic, A. M. Tekalp, "Maximum likelihood parametric blur identification based on a continuous spatial domain model", *IEEE Transactions on Image Processing*, vol. 1, no. 4, pp. 496-504, Oct. 1992.

[74] J. H. Elder and S. W. Zucker, "Local scale control for edge detection and blur estimation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 7, Jul. 1998.

[75] M. Basu, "Gaussian-based edge-detection methods-a survey", *IEEE Transactions on Systems, Man, and Cybernetics: Applications and Reviews*, vol. 32, no. 3, pp. 252- 260, Aug. 2002.

[76] K. Suzuki, I. Horiba and N. Sugie, "Neural edge enhancer for supervised edge enhancement from noisy images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1582-1596, Dec. 2003.

[77] R. L. Lagendijk, J. Biemond and D. E. Boekee, "Identification and restoration of noisy blurred image using the expectation-maximization algorithm", *IEEE Transactions on Acoustic, Speech and Signal Processing*, vol. 38, pp. 1181-1191, 1990.

[78] S. K. Kim, S. R. Park, and J. K. Paik, "Simultaneous out-of-focus blur estimation and restoration for digital auto-focusing system", *IEEE Transactions on Consumer Electronics*, vol. 34, pp. 1071-1075, 1998

[79] A. K. Katsaggelos, "Iterative image restoration algorithms", *Optical Engineering*, vol.287, pp. 735-748, 1989.

[80] M. Zhao, M. Bosma, and G. de Haan, "Making the best of legacy video on modern displays", *Journal of the Society for Information Display*, vol. 15, no. 1, pp. 49-60, Jan. 2007.

[81] M. Zhao, R. E. J. Kneepkens, P. M. Hofman and G. de Haan, "Content adaptive image de-blocking", *IEEE International Symposium on Consumer Electronics*, pp. 299-304, Sept. 2004.

[82] L. L. Scharf, *Statistical signal processing*, Addison-Wesley, Reading, MA, 1991, ISBN 0201190389.

[83] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm". *Journal of the Royal Statistical Society*, Series B, vol. 39, pp. 1-38, 1977.

[84] J. A. Hartigan and M. A. Wong, "A K-means clustering algorithm", *Applied Statistics*, vol. 28, no. 1, pp.100-108, 1979.

[85] J. W. Tukey, "Nonlinear methods for smoothing data", *Conf.Rec.*, Eascon, pp. 673, 1974.

[86] J. S. Lee, "Digital image smoothing and the sigma filter", *Computer Vision, Graphics and Image Processing*, vol. 24, pp. 225-269, 1983.

[87] D. R. K. Brownrigg, "The weighted median filter", *Commun. Assoc. Comput. Mach.*, vol. 27, no. 8, pp. 807-818, Aug. 1984.

[88] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: A tutorial", *IEEE Trans. Circuits Syst.*, vol. 41, pp. 157-192, May 1996.

[89] L. Yin, J. Astola, and Y. Neuvo, "A new class of nonlinear filters neural filters", *IEEE Trans. Signal Processing*, vol. 41, no. 3, pp. 1201-1222, Mar. 1993.

[90] A. C. Bovik, T. S. Huang, and D. C. Munson, "A generalization of median filtering using linear combinations of order statistics", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1342-1349, Dec. 1983

[91] Y. Lee and S. Kassam, "Generalized median filtering and related nonlinear filtering techniques", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 31, pp. 672-683, Jun. 1985

[92] P. P. Gandhi and S. A. Kassam, "Design and performance of combination filters for signal restoration", *IEEE Trans. Signal Processing*, vol. 39, pp. 1524-1540, Jul. 1991.

[93] F. Palmieri and C. G. Boncelet, Jr., "Ll-filters-a new class of order statistic filters", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 691-701, May 1989

[94] K. E. Barner and G. R. Arce, "Permutation filters: A class of nonlinear filters based on set permutations", *IEEE Trans. Signal Processing*, vol.42, pp. 782-798, Apr. 1994.

[95] R. C. Hardie and K. E. Barner, "Extended permutation filters and their application to edge enhancement", *IEEE Trans. Image Processing*, vol. 5, pp. 855-867, Jun. 1996.

[96] E. A. Thompson, R. C. Hardie and K. E. Barner, "Hybrid order statistic filter and its application to image restoration", *Applied Optics*, vol. 40, pp. 656-661, Feb. 2001.

[97] T. Kondo, Y. Fujimori, S. Ghosal and J. J. Carrig, " Method and apparatus for adaptive filter tap selection according to a class", US-Patent: US 6,192,161 B1, Feb. 20, 2001.

[98] T. Kondo, Y. Okumura, and Y. Node, "Picture conversion apparatus picture conversion method learning apparatus and learning method", US-Patent: US 6,323,905 B1, Nov. 27, 2001.

[99] Y. Nie, and K. E. Barner, "Fuzzy transformation and its applications", *IEEE International Conference on Image Processing*, Barcelona, Spain, Sept., 2003

[100] Y. Nie, and K. E. Barner, "Optimized fuzzy transformation for image deblocking", *IEEE International Conference on Multimedia and Expo*, Baltimore, Maryland, July, 2003

[101] H. R. Wu, "A new distortion measure for video coding blocking artifacts", *Proceedings of IEEE International Conference on Communication Technology*, vol. 2, pp. 658-661, May, 1996.

[102] X. Li and M. T. Orchard, "New edge-directed interpolation". *IEEE Transactions on Image Processing*, vol. 10, no. 10, Oct. 2001, pp. 1521-1527.

[103] H. Y. Lee, J. W. Park, T. M. Bae, S. U. Choi, and Y. H. Ha, "Adaptive scan rate up-conversion system based on human visual characteristics", *IEEE Trans. on Consumer Electronics*, vol. 46, no. 4, pp. 999-1006, Nov. 2000.

[104] G. de Haan and R. Lodder, "De-interlacing of video data using motion vectors and edge information", *Digest of the ICCE 2002*, pp. 70-71, Jun. 2002.

[105] C. B. Atkins, C. A. Bouman, and J. P. Allebach, "Optimal image scaling using pixel classification", *International Conference on Image Processing*, vol.3, pp864-867, Oct. 2001.

[106] N. Plaziac, "Image interpolation using neural networks", *IEEE Trans. Image Processing*, vol. 8, no. 11, pp. 381-390, 1999.

[107] R. Rojas, *Neural Networks - A Systematic Introduction*, Springer-Verlag, Berlin, New-York, 1996. ISBN 3540605053.

[108] J. Go, K. Sohn and C. Lee, "Interpolation using neural networks for digital still cameras", *IEEE Trans. Consumer Electronics*, vol. 46, no. 3, pp. 610-616, Aug. 2000.

[109] C. Staelin, D. Greig, M. Fischer and R. Maurer, "Neural network image scaling using spatial errors", HP Technical Report, HPL-2003-26R1.

[110] M. Zhao, J. A. Leitao and G. de Haan, "Towards an overview of spatial up-conversion techniques", *Proceedings of ISCE'02*, pp. E13-E16, Sep. 2002.

[111] R. Fletcher, *Practical methods of optimization*, Second Edition, John Wiley & Sons Ltd, pp. 100-107, Aug. 2000. ISBN 0471494631.

[112] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function", *IEEE Trans. Information Theory*, vol. 39, no. 3, pp. 930-945, Mar. 1993.

[113] A. K. Jain, *Fundamentals of digital image processing*, Englewood Cliffs, NJ: Prentice-Hall, 1989, ISBN 0133361659.

[114] W. K. Pratt, *Digital image processing*, New York: Wiley, 1978, ISBN 0471018880.

[115] R. C. Gonzalez and R. E. Woods, *Digital image processing*, Addison-Wesley Publishing Company, Reading, MA, 1993, ISBN 0201508036.

[116] D. J. Ketcham, R. Lowe, and W. Weber, "Seminar on image processing", *Real-Time Enhancement Techniques*, pp. 1-6, 1976.

[117] R. Hummel, "Image enhancement by histogram transformation", *Computer Graphics Image Processing*, vol. 6, pp. 184-195, 1977.

[118] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. H. Romeny, J. B. Zimmerman, and K. Zuiderveld, "Adaptive histogram equalization and its variations", *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 355-368, 1987.

[119] S. M. Pizer, R. E. Johnston, J. P. Ericksen, B. C. Yankaskas, and K. E. Muller, "Contrast-limited adaptive histogram equalization: speed and effectiveness", *Proceedings of Conference on Visualization in Biomedical Computing*, pp. 337-345, 1990.

[120] V. T. Tom and G. J. Wolfe, "Adaptive histogram equalization and its applications", *SPIE Application on Digital Image Processing IV*, vol. 359, pp. 204-209, 1982.

[121] K. Zuiderveld, "Contrast limited adaptive histogram equalization", *Graphics Gems IV*, Cambridge, MA, Academic Press, pp. 474-485, 1994.

[122] S. Bae, S. Paris and F. Durand, "Two-scale tone management for photographic look", *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 637-645, 2006.

[123] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity", *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004.

[124] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences", *Proceedings of the 15th International Conference on Machine Learning*, 1998.

[125] J. P. Vink and G. de Haan, "Local estimation of video compression artifacts", *IEEE International Conference on Consumer Electronics*, Las Vegas, USA, Jan. 2009.

[126] H. Hu, P. M. Hofman and G. de Haan, "Content-adaptive neural filters for image interpolation using pixel classification", *Proceedings of SPIE, Applications of Neural Networks and Machine Learning in Image Processing conference IX*, San Jose, Jan. 2005.

[127] H. Hu and G. de Haan, "Classification-based hybrid filters for image processing", *Proceedings of SPIE, Visual Communications and Image Processing*, San Jose, USA, Jan. 2006.

[128] H. Hu and G. de Haan, "Low cost robust blur estimator", *Proceedings of IEEE International Conference on Image Processing*, Atlanta, USA, pp. 617-620, Oct. 2006.

[129] H. Hu and G. de Haan, "Simultaneous coding artifacts reduction and sharpness enhancement", *Proceedings of IEEE International Conference on Consumer Electronics*, Las Vegas, USA, Jan. 2007.

[130] H. Hu and G. de Haan, "Trained bilateral filters and applications to coding artifacts reduction", *Proceedings of IEEE International Conference on Image Processing*, San Antonio, pp. 325-328, Sept. 2007.

[131] L. Shao, H. Hu and G. de Haan, "Coding artifacts robust resolution up-conversion", *Proceedings of IEEE International Conference on Image Processing 2007*, vol. 5, pp. 409-412, Sept. 2007.

[132] H. Hu and G. de Haan, "Adaptive image restoration based on local robust blur estimation", *Proceeding of the 9th International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 461-472, Volume LNCS 4678, Springer, Aug. 2007.

[133] H. Hu and G. de Haan, "Class-count reduction techniques for content adaptive filtering", *Proceedings of The 13th IEEE International Symposium on Consumer Electronics*, Kyoto, Japan, pp. 11-15, May 2009.

[134] G. de Haan and H. Hu, "Simultaneous coding artifact reduction and sharpness enhancement", European Patent No. EP06301213, Dec. 2006.

[135] J. P. Vink, H. Hu and G. de Haan, "Local spatially consistent metric for image artefacts", European Patent No. EP08167011, May 2008.

[136] G. de Haan and H. Hu, "Image data processing and arrangements therefor", WO Patent No. WO/2009/034555, Sept. 2008.

Curriculum Vitae

Hao Hu was born in Qichun, Hubei Province, P. R. China, on January 10th, 1980. He received the B.Eng degree in Electrical Engineering from Huazhong University of Science and Technology, Wuhan, P. R. China in 2002. Then he joined the Design Technology Institute, a joint institute between National University of Singapore and Eindhoven University of Technology and received the Master of Technological Design (MTD) degree in Embedded Systems in 2004. From 2005 to 2008 he was working towards the Ph.D. degree at the department of Electrical Engineering at Eindhoven University of Technology on the subject of content adaptive video enhancement. He joined Philips Research Eindhoven in January 2009 where he is currently a Research Scientist in the Video Processing and Analysis group.