Image and Video Resolution Enhancement Using Sparsity Constraints and Bilateral Total Variation Filter

IMAGE AND VIDEO RESOLUTION ENHANCEMENT USING SPARSITY CONSTRAINTS AND BILATERAL TOTAL VARIATION FILTER

BY

ZAHRA ASHOURI TALOUKI, B.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

Master of Applied Science (2012) McMaster University

(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE:	Image and Video Resolution Enhancement Using Sparsity Constraints and Bilateral Total Variation Filter
AUTHOR:	Zahra Ashouri Talouki B.Sc., (Electrical Engineering) Isfahan University of Technology, Isfahan, Iran
SUPERVISOR:	Dr. Shahram Shirani
NUMBER OF PAGES:	xiv, 64

To my dear parents and beloved husband

Abstract

In this thesis we present new methods for image and video super resolution and video deinterlacing. For image super resolution a new approach for finding a High Resolution (HR) image from a single Low Resolution (LR) image has been introduced. We have done this by employing Compressive Sensing (CS) theory. In CS framework images are assumed to be sparse in a transform domain such as wavelets or contourlets. Using this fact we have developed an approach in which the contourlet domain is considered as the transform domain and a CS algorithm is used to find the high resolution image.

Following that, we extend our image super resolution scheme to video super resolution. Our video super resolution method has two steps, the first step consists of our image super resolution method which is applied on each frame separately. Then a post processing step is performed on estimated outputs to increase the video quality. The post processing step consists of a deblurring and a Bilateral Total Variation (BTV) filtering for increasing the video consistency. Experimental results show significant improvement over existing image and video super resolution methods both objectively and subjectively.

For video deinterlacing problem a method has been proposed which is also a two step approach. At first 6 interpolators are applied to each missing line and the

interpolator which gives the minimum error is selected. An initial deinterlaced frame is constructed using selected interpolator. In the next step this initial deinterlaced frame is fed into a post processing step. The post processing step is a modified version of 2-D Bilateral Total Variation filter. The proposed deinterlacing technique outperforms many existing deinterlacing algorithms.

Acknowledgements

I would like to express my profound gratitude to my supervisor, Dr.Shahram Shirani who has supported me throughout my research and thesis writing with his patience and knowledge. He provided encouragement, support and sound advice.

Besides I would like to thank committee members, Dr. Shahin Sirouspour and Dr. alexandru patriciu for taking the time to read my thesis and for providing me with great comments.

Last but not the least, I would like to thank my parents and my husband for their love and support, without which I could not have been at this stage in my life.

Notation

Abbrevation

2-D	Two Dimensional
3-D	Three Dimensional
4FLMC	4 Filed Motion Compensation
BTV	Bilateral Total Variation
CIF	Common Intermediate Format
CS	Compressive Sensing
DCT	discrete cosine transform
DFB	Directional Filter Bank
DIMC	directional interpolation and motion compensation
DSP	Digital Signal Processing
FAV	Field Averaging
FI	Field Insertion
HMA	hierarchical motion analysis

HVS	human visual system
IBP	Iterative Back-Projection
LA	Line Averaging
LP	Laplacian Pyramid
LR	Low Resolution
MA	Motion Adaptive
MC	Motion Compensated
MCAMA	Motion Compensation Aided Motion Adaptive
ME	Motion Estimation
MOMA	Motion Compensation Assisted Motion Adaptive
MRF	Markov Random Field
NC	Not Considered
NNDMF	Neural Network Deinterlacing Using Multiple Fields
OMP	Prthogonal Matching Pursuit
POCS	Projection onto Convex Sets
PSNR	Peak Signal to Noise Ratio
QCIF	Quarter Common Intermediate Format
RIC	Restricted Isometry Condition

ROMP	Regularized Orthogonal Marching Pursuit
SAD	Sum of Absolute Difference
SR	Super Resolution
TMCD	True Motion-Compensated De-Interlacing
VT	Vertical Temporal

Contents

List of Figures

Chapter 1

Introduction

1.1 Image Super Resolution

Super Resolution (SR) is a technique to make a high resolution image out of one or a set of Low Resolution (LR) frames. In fact in low resolution images the high frequency components are missing and SR tries to estimate those missing frequencies in a way that the difference between the original image and the reconstructed SR one is minimum [1].

There are different methods to generate High Resolution (HR) images from the LR ones. Some methods used the reconstruction based approaches and some others use learning algorithms. Reconstruction based approaches use the information in separate frames for solving the problem of SR. In fact these methods use the independent information in different frames and combine them to make the HR image. In order to have independent information in different frames it is mandatory for samples of LR images to be sub-pixel shifted from samples of other LR images. Otherwise there is no independent information available [2]. Projection onto Convex Sets (POCS) [3]

and Iterative Back-Projection (IBP) Methods [4] are among the famous SR methods in literature.

Learning based approaches use a specific image sequence set to learn their characteristics and use those known characteristics as extra information to reconstruct the HR image. A contourlet method was proposed by C. V. Jiji and S. Chaudhuri which uses learning concepts [5]. In this work a dictionary of high resolution images and their contourlet coefficients is available. Then for each LR image contourlet transform is computed with levels of decomposition being 1 level less than the levels of decomposition of dictionary. Finally for reconstructing HR image coefficients of that extra level will be found by using the dictionary.

Recently there are some new methods introduced which only use a single LR image for finding the HR image. This is an ill-posed problem and requires more conditions to solve the problem.

Image interpolation is a solution for generating a HR image from the associated LR capture. Its applications are in medical imaging and up conversion of standard definition video frames. Some algorithms like bilinear and bicubic are easy and fast in implementation but they fail to capture the varying pixel structure around edges, as a result the picture would be blurred and with ringing artifacts around edges [6].

With the increasing demand for higher resolution images more powerful methods for interpolation were proposed to preserve edge sharpness and produce images with fewer artifacts. An edge directed interpolation was presented by Li and Orchard where the authors estimate the covariance of the HR image from the variance of the LR image [7]. Zhang and Wu proposed a method in which a missing pixel was interpolated in

multiple directions and the results were fused by minimum mean square-error estimation [8]. An up scaling approach for resizing images in the discrete cosine transform (DCT) transform domain was proposed by Park, et al [9]. The paper uses the property that the multiplication in spatial domain corresponds to the symmetric convolution in DCT domain.

P. Sen and S. Darabi use the sparsity of an image in the wavelet domain and CS theory to interpolate the HR image [10]. The sparse mixing estimator method [11] proposed a new class of adaptive estimators acquired by mixing a family of linear inverse estimators which are derived from different priors on the signal regularity. S. Yang, et al use local gradient features to propose an interpolation method which preserves edge sharpeness [12]. N. Mueller, et al used wavelet and contourlet transform and a simple thresholding step to find the largest contourlet coefficients and reconstruct the HR image [13]. Another method was presented in [14] where the author uses a hybrid method which combines the simplicity of ML and some prior knowledge. Also in [15] an algorithm based on MRF is used for SR.

1.2 Video Super Resolution

A generalization of the image interpolation is image sequences interpolation which is referred as video resolution enhancement. In video resolution enhancement our goal is to produce a HR version of a given video stream in a manner that is visually pleasing and artifact-free. According to the type of information used video resolution enhancement can be divided in two groups. In the first group the algorithm only uses the information available in the current frame, this group which is very similar to image super resolution is called spatial methods. The second group also uses the

information from neighboring frames which causes higher quality and is referred to as spatiotemporal methods.

H.Takeda et al introduced a method based on multidimensional kernel regression, where each pixel in the video sequence is approximated with a 3-D local (Taylor) series, capturing the essential local behavior of its spatio-temporal neighborhood [16]. Also in motion assisted steering kernel (MASK) paper [17] a 3-D steering kernel regression approach has been proposed. The paper suggests an algorithm for multi-frame video interpolation and denoising. X. Zhang, et al, propose a robust video super-resolution method [18] considering temporal information reliability with registration efficiency model following by a spatial-temporal steering kernel.

Method proposed in [19] is a novel framework for adaptive enhancement and spatiotemporal up scaling of videos which contain complex activities without the explicit need for motion estimation. Z. Wu, et al, proposed a new hybrid DCT-Wiener based interpolation [20]; where the authors take advantage of interpolation in both spatial and DCT domain and fuses this two approaches to design an improved up sampling method. An implementation of a super resolution algorithm for video sequences based on a Digital Signal Processor (DSP) is proposed by S. Lopez, et al [21], where the method is used for real time consumer applications. A number of fast block matching algorithms are compared by G. M. Callico, et al [22], to select the one which presents the best tradeoff between the computational load and super resolved video quality.

1.3 Video Deinterlacing

The interlaced scan method has been the dominant scan format for recording, broadcasting, storing and displaying of video and television standards such as NTSC, PAL,

and SECAM for the efficient usage of limited bandwidth. Interlacing is cost efficient and has until recently been sufficient to ensure the best possible viewing experience to the human visual system (HVS). Interlacing is still used for most standard definition TVs, but not for LCD, micro mirror (DLP), or plasma displays; these displays do not use a raster scan to create an image, and so cannot benefit from interlacing. Interlacing artifacts that were not a problem in former TV's have started to become visible as screens have grown larger and brighter with higher contrasts [23][24]. The conversion of interlaced video to progressive video is referred to as de-interlacing. De-interlacing convert each field into one frame. The number of pictures per second remains constant in this conversion but the number of lines in each picture is doubled. De-interlacing can add to the cost of a television set using such displays. Currently, progressive displays dominate the HDTV market.

A set of lines that describe a picture together are called frame. A frame consists of two fields, odd field (also known as the top field) and even field (also known as the bottom field). In fact de-interlacing is the process to convert field data into frame data. Frame data contains all the lines of an image while field data only contains either even-indexed or odd-indexed lines. The process of deinterlacing is shown in Fig. 1.1. This conversion leads to various visual defects which the de-interlacing process should try to avoid [23][25].

De-interlacing is a simple task for stationary pictures, in which there is no object or camera motion or intensity change. In these circumstances the alternating odd and even fields described the captured scene. However most of the time this is not the case; in fact objects move, the camera moves, the light changes and also scene cuts frequently occur. So de-interlacing requires the interpolation of picture data that was

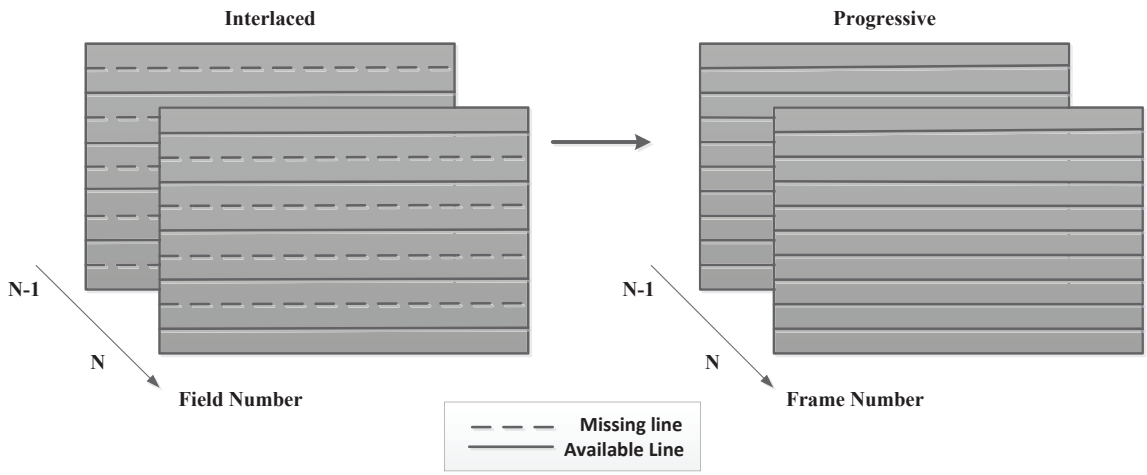


Figure 1.1: The process of deinterlacing

never transmitted or even registered. As de-interlacing is a simple matter for stationary image parts, we can create virtually stationary images by compensating true motion. In motion compensation we do need motion estimation. Motion estimation (ME) was and is still subject to much research and is a fundamental problem in many video processing tasks [26].

Over the last two decades, many de-interlacing algorithms have been proposed. These algorithms range from simple spatial interpolation, via directional dependent filtering, up to advanced motion-compensated (MC) interpolation.

Spatial interpolation techniques are among simple methods which uses the information in above and below known lines. Generally spatial methods only interpolate within a single field, rather than merging information from adjacent fields, to preserve the smoothness of motion. The simplest way to double lines is to repeat the above known line. This method along with Line Averaging (LA) or 'bob', which is

the fifty-fifty average of the above and below line, has been widely used in practice. The edge directional interpolation techniques such as Edge-based LA (ELA), Efficient ELA (E-ELA), Modified ELA (M-ELA), and Direction-Oriented Interpolation (DOI) perform interpolation in the direction of the highest correlation among pixels. In fact edge-directed algorithms such as [27] and [28] estimate the dominant (spatiotemporal) edge direction and then avoid interpolating across the edge.

Temporal methods use the information in adjacent frames. Field Insertion (FI) is one of the simplest temporal deinterlacer. It is named as merging or weaving as well. In this method the unknown lines are filled by repeating the neighbouring lines in time. Field Averaging (FAV) is another method among this group which averages the before and after temporal neighbouring lines of a blank line. Also the Vertical Temporal Interpolation (VT) is a fifty-fifty combination of LAV and FAV. The VT interpolator combines the benefits from the pure spatial and temporal interpolators. As interpreted from the spatial and temporal explanations, it is clear that in static regions temporal interpolation will yield in accurate result while when motion exists spatial interpolation is preferred. To overcome such problems Motion Adaptive Deinterlacing (MA) can be applied. Motion adaptation algorithms decide whether to interpolate spatially or temporally based on the results of a motion detection algorithm. If the scene is moving, spatial interpolation is used to avoid mouth teeth artifacts. If the scene is stationary, temporal interpolation is used to obtain the highest possible resolution.

Motion adaptive de-interlacing is among non-linear techniques for interpolating. To detect motion, the difference between two pictures is calculated, due to noise this signal is not equal to zero in the static parts of the picture. Some systems may have

some other problems as well like timing jitter of the sampling clock. One way to get rid of most of the interlacing artifacts is to use motion adaptive deinterlacers. In these kinds of motion deinterlacers, motion detection is used to determine whether to use local temporal interpolation or use spatial interpolation in the case of motion. Since the human visual system is highly sensitive to edges, even a single edge which is poorly deinterlaced can significantly lower the visual quality of the progressive output. This problem can be handled by smoothing the problem region. But the blurring problem is an artifact that will still remain and annoys the viewer. To overcome this problem Motion Compensated (MC) deinterlacing methods are introduced. Motion compensation allows us to virtually convert a moving sequence into a stationary one. The motion compensated de-interlacers, clearly exceed both the linear and non-linear de-interlacers in image quality, but require significant compute resources to implement in real time. The MC Field insertion or zero-order MC temporal interpolation implies that the original samples of a field are shifted versions of the previous field over a one field period. In MC Field averaging or the first order MC temporal interpolation, both the previous and next field are considered into account. The orders can grow large, in this case some problems may occur for example the validity of motion vectors are reduced as the order increases and also there is an increasing need for storage.

Temporal backward projection extends the motion vector to the pre-previous field if the desired pixel is not in the previous field. In fact if the motion vector does not point to a special pixel in the previous field it goes further to find it in the pre-previous field. If the motion is in none of them, then intra-field (interpolation in the field without considering other fields) interpolation is applied. One disadvantage of

this method is that the motion is considered to be uniform implicitly, which might not be true. Also the algorithm is not efficient for the incorrect motion vectors. In the variational method explained in [23], an energy function based on the spatial distribution of intensities and temporal distribution of intensities is introduced. The optimum solution is considered to be the one that gives the minimum energy.

An alternative approach is proposed in [25] where the deinterlacing problem is posed as a Markov random field (MRF). In this case a data cost is used to encourage the missing pixels to be close to their neighbors in the rows above, below, before, and after. Another method was proposed in [29] where the benefits of rule-based algorithms such as motion-adaptation, edge-directed interpolation, and motion compensation are combined with those of an MRF formulation. H. J. Cho, et al proposed a voting based deinterlacing method for directional error correction (VDD) [30], which can reduce wrong edge direction.

1.4 Motivation and Contribution

One of the problems which affect the quality of an interpolation method is sharpness of edges. Several authors have tried to solve this problem with different methods but still in most of the interpolation methods, edges are blurred with ringing artifacts. In our image and video interpolation approach we tried to find a solution for the jagginess problem of the edges.

We have used the contourlet transform for this goal. Contourlet transform is a transform which changes the input signal to a sparse output. In fact the output contourlet coefficients are a sparse representation of the input image. Special features of contourlet transform enable us to detect edges and reconstruct them with fewer artifacts.

We also incorporate the newly introduced field of compressive sensing in our image interpolation method. Compressive sensing introduces some techniques to solve underdetermined problems. Our contribution in the image super resolution step is using sparsity in the contourlet domain along with CS framework to reconstruct an image. We then generalize our method to video resolution enhancement. Our contribution in this area is adding a 3-D filter for quality improvement. We have generalized the 2-D Bilateral Total Variation (BTV) filter introduced in [1] to a 3-D BTV filter to be used for filtering video sequences which are 3-D data. In fact after applying our image interpolation technique to each frame we apply a 3-D BTV filter to the whole sequence. We also have proposed a method for video deinterlacing which is based on a two step algorithm. In the first step 6 interpolators are used and among these interpolators the one which gives minimum error is chosen for each window in missing line where the window size is adaptive. In the next step a modified version of the 2-D BTV filter is applied as a post processing step. Our contribution for video deinterlacing is mostly in the second step which is modifying the 2-D BTV filter in a way to make it applicable to the deinterlacing problem.

1.5 Wavelet Transform

As a mathematical tool, wavelets can be used to extract information from different kinds of data, including audio signals and images. Mallat was the first one who applies wavelets to multiresolution analysis. Multiresolution theory is basically representing a signal in more than one resolution. Wavelets can be used as basis for signal expansion.

A given signal x can be expressed as linear combination of wavelets $\{\psi\}_n$.

$$x = \sum_{n=1}^{\infty} c_n \psi_n \quad where \quad c_n = \langle x, \psi_n \rangle \tag{1.1}$$

The wavelet coefficients are sparse, moreover fast implementations of wavelet transform are available. For these reasons wavelets have become a useful tool in signal processing applications especially one-dimension cases [31]. In the two-dimension case the story is somehow different. The 2-D image is decomposed using a two stage filter bank where one stage performs 1-D filtering on the rows and the second stage performs it on the columns as shown in Fig. 1.2. In fact the 2-D wavelets are constructed by outer product of 1-D wavelets. As a result there are inefficiencies, for example the 2-D wavelet can only use squares made by the tensor product for capturing the contour. It is clear that it would cause problem in finer scales. Also it can only capture the edges which are in vertical, horizontal and diagonal directions. In order to solve the first problem curvelets were introduced. Curvelets have rectangular shape for capturing contours and this will help to capture the contour much faster and with less error [32]. On the other hand contourlets help to solve the problem of directions.

1.6 Contourlet Transform

Contourlet is a generalization of wavelet in a way that instead of having just 3 directions it can pick as many directions as needed. This property enables contourlet to capture edges better and sharper than wavelets.

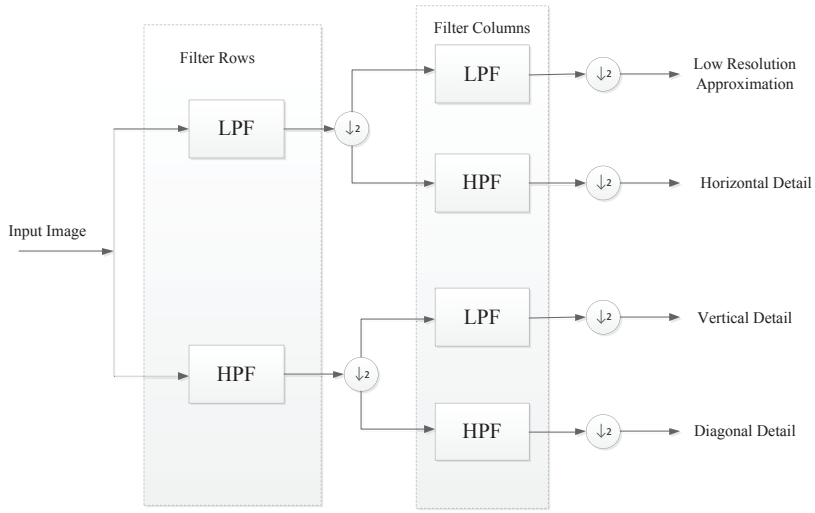


Figure 1.2: Single level 2-D Wavelet Transform

Contourlet uses Directional Filter Bank (DFB) to decompose the signal into several directions in each resolution level. Discrete Contourlet Transform is a combination of Laplacian Pyramid (LP) and Directional Filter Bank. Its Structure is shown in Fig. 1.3. In the first stage Laplacian Pyramid is applied to the input image and the image is partitioned into low frequency components and high frequency components. After each level of decomposition, high frequency part is fed to a DFB and it is partitioned to several directions. In fact it predicts the original image using low frequency components and computes the difference between the original image and prediction, which is a band-pass image. This difference forms the input to the DFB. In images pixels near each other are correlated. This correlation causes the direct representation of an image very inefficient. Laplacian Pyramid is a technique to de-correlate neighboring pixels. The subtraction of the prediction from the original image de-correlates pixel values and hence the representation of this difference is less

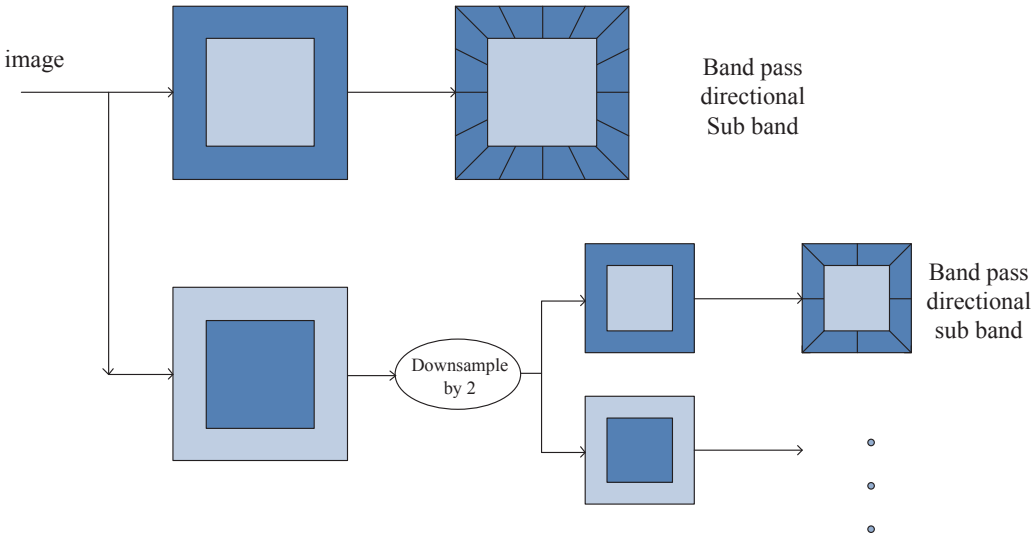


Figure 1.3: Contourlet transform using directional filter bank.

redundant [33].

The second part in contourlet transform is Directional Filter Bank. A construction of 2-D directional filter bank is given in [34] by Bamberger and Smith, on which the DFB is implemented by an l -level binary tree decomposition which causes to have 2^l subbands. In this approach in order to find the finer directional subbands, a complicated tree expanding rule should be followed. A simpler approach was proposed in [31]. This method is basically constructed of two steps. The first step is realized by a two-channel quincunx filter bank with fan filters which divide a 2-D signal into horizontal and vertical direction. In Fig. 1.4 two levels of directional filter bank is shown. The filters in this figure are fan filters and Q_0 and Q_1 are quincunx sampling matrices. The second step is a shearing operator. The shearing (rotating) operator only changes the order of data and rearranges it.

Combination of Laplacian Pyramid and Directional Filter bank would form contourlet

transform or *multi-scale and directional decomposition*. By iterating on the coarser versions of the image, levels of decomposition can be generated. Usually the number of directions will reduce in coarser versions of the image, since there is not much detail on them [32][31]. Specifically if we consider $a_0[n]$ as the input image, the output of LP stage is $b_j[n]$, $j = 1, 2, ..., J$ (from fine to coarse version) which are the bandpass images and the low pass image $a_J[n]$. In the next stage each bandpass image $b_j[n]$, is decomposed by an l_j-level DFB into 2^{l_j} bandpass directional images.

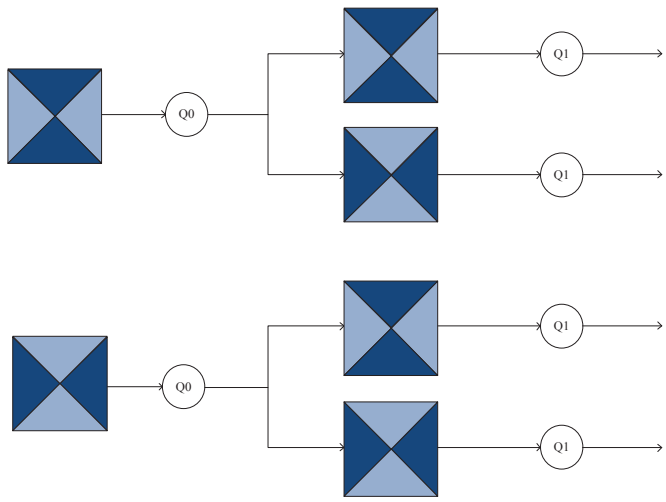


Figure 1.4: First two levels of DFB.

1.7 Compressed Sensing

Compressed sensing, also known as compressive sensing (CS) is a framework for reconstructing signals that are considered to be sparse or compressible. Assume a signal that is represented as an N-dimensional real vector. If this vector is sparse in some

linear basis, compressive sensing can reconstruct signal accurately using only a small number of basis-function coefficients. In fact by using compressive sensing one can measure an N-dimensional signal not directly but with a new set of related measurements that are typically much smaller. This potentially simplifies the sensing system [35].

Shannon-Nyquist sampling theorem indicates that a signal must be sampled at the rate of 2 times the highest frequency in the signal to be able to recover it without losing any part of signal. By CS it is possible to achieve sampling rates which are much smaller than the minimum sampling rate introduced by *shannon* without losing data. Assume signal S is of length L with K ($K \ll L$) non-zero coefficients in some linear basis (S is a sparse signal). By using compressive sensing methods we can reconstruct signal S ($\widehat{S}$) if we only have those K non-zero coefficients.

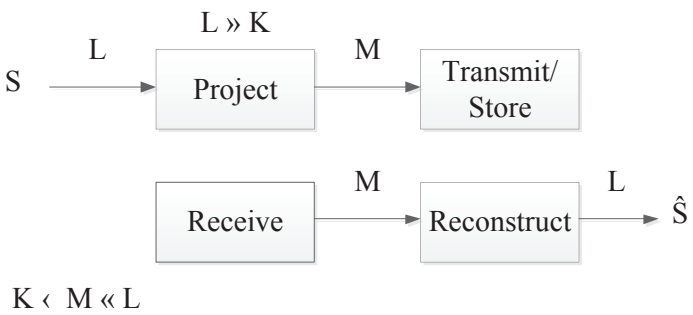


Figure 1.5: Projection and reconstruction of a signal using sparse transform.

Fig. 1.5 shows a block diagram of a CS system. In the first block signal is projected onto a sparse transform and M coefficients ($K < M \ll L$) are chosen (Vector U in Fig.1.6). The sparse transform can be wavelets, contourlets or other

sparse representations. Fig. 1.6 basically shows what happens in the reconstruction block in Fig. 1.5. In this part algorithm tries to find signal S which is K-sparse (meaning to have K non-zero coefficients) by having Matrix B and M coefficients (vector U).

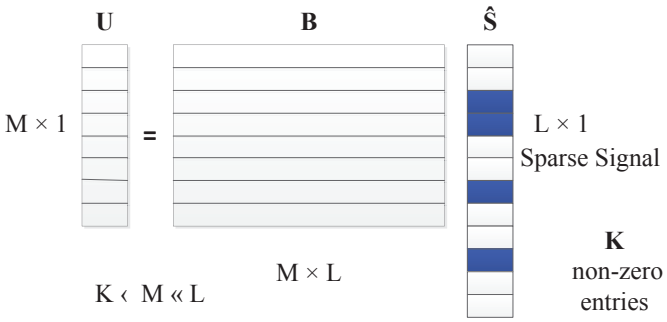


Figure 1.6: Replace samples with few linear projections.

Matrix B in Fig. 1.6 is the measurement matrix. Measurement matrix is a matrix which allows reconstruction of signal with length L from M measurements (Vector U). The process of finding $\widehat{S}$ from M coefficients (U) is explained more in chapter 2. In image processing the main assumption is that the image has a sparse representation in an orthonormal basis or a tight frame. This assumption will help solving the underdetermined problems which have more unknowns than equations [10]. In fact compressive sensing take advantage of sparsity in a particular domain on which many coefficients are close to or equal to zero.

Orthogonal Matching Pursuit (OMP) and *Regularized Orthogonal Matching Pursuit* (ROMP) are among greedy algorithms for solving underdetermined problems. These methods are explained in the following chapter.

Chapter 2

Image Interpolation Using Sparsity in Contourlet Domain

In our image interpolation method, we have used the newly introduced field of Compressed Sensing (CS) for reconstruction. The main contributions are:

1) the problem of interpolation is posed in the CS frame work

2) the image is considered to be sparse in the transform domain which in this work is contourlet domain.

The problem of finding a HR image out of a single LR image is an underdetermined problem, since the number of pixels in HR (unknowns) is more than the number of pixels in LR. Consider x to be the vectorized version of the HR image. Our goal is to find x from the vectorized LR image $\widehat{x}$ such that:

$$\widehat{x} = Ax \tag{2.1}$$

where A is the sampling operator and $\widehat{x}$ and x are of length m and n respectively with $m << n$. Clearly this problem does not have a unique solution. Signal x can be considered to be k-sparse in some transform domain, meaning to have k non-zero coefficients in that domain. So equation (2.1) would be modified to:

$$\widehat{x} = A\psi\widetilde{x} = \varphi\widetilde{x} \tag{2.2}$$

where $A\psi = \varphi$ and $\widetilde{x}$ is the vector of coefficients in the transform domain which is k-sparse under some basis ψ .

Now the problem would be to find the $\widetilde{x}$ coefficients while having $\widehat{x}$. Once we do have the coefficients, the HR image x could be found by applying inverse of the transform [13].

To have a unique result in this underdetermined equation, $\widetilde{x}$ is considered to have a minimum number of non-zero coefficients. In other word between all possible solutions for $\widetilde{x}$ the one which gives the minimum number of coefficients is chosen. In mathematical word the desired $\widetilde{x}$ can be found by solving an l_0-optimization problem. Greedy algorithms given by CS researchers provide a solution for l_0-norm.

$$min\|\widetilde{x}\|_0 \quad s.t \quad \widehat{x} = \varphi\widetilde{x} \tag{2.3}$$

2.1 Orthogonal Matching Pursuit (OMP) and Regularized OMP

In the OMP method at first the transform is applied on the input and then the largest coefficient is chosen. This coefficient is assumed to be the only non-zero coefficient

and based on it a least-square problem is solved to find an estimate of the desired output. Then the estimated output is subtracted from the original LR image and the algorithm will iterate on this difference, each time just choosing the largest coefficients until the desired number of non-zero coefficients is found.

In this method $\widehat{x}$ and the measurement matrix φ are given. We then can find the coefficients of $\widetilde{x}$ by computing $\varphi^*\widehat{x}$ (φ^* is the backward matrix of φ) and selecting the largest coefficient. This coefficient is assumed to be the only non-zero coefficient and a least-square problem is solved based on it to find an estimate of $\widehat{x}$. Then the estimated output is subtracted from the original LR image and the algorithm will iterate on this difference until the desired number of non-zero coefficients is found.

Although OMP is a very efficient algorithm, a modified version of it called Regularized OMP (ROMP) was proposed that has superiority over OMP. ROMP is faster; also it is more robust to Restricted Isometry Condition (described next).

2.2 Restricted Isometry Condition (RIC)

The measurement matrix φ satisfies RIC with parameters (k, ε) if for a constant number of $\varepsilon \in (0, 1)$ we have:

$$(1-\varepsilon)\|v\|_2 \leq \|\varphi v\|_2 \leq (1+\varepsilon)\|v\|_2 \tag{2.4}$$

with every k-sparse vector v [10].

In fact this condition implies that an acceptable measurement matrix φ is a matrix in which each set of k columns forms an almost orthogonal set. In order to have such a matrix in our case we do need matrices A and ψ to be as incoherent as possible [36].

In our problem A is the point sampling matrix and ψ is the contourlet transform. But point sampling operator is coherent with bases which are good at representing localized features because in these bases, small spatial features (e.g. point samples) can be represented with only a few coefficients.

2.3 Proposed Approach

As described earlier, ψ in equation (2.2) is the notation for transform basis and $\widetilde{x}$ is the vector of coefficients in the transform domain. In this work the transform is considered to be contourlet transform. Now the problem would be to find the $\widetilde{x}$ coefficients with having $\widehat{x}$. Once we do have the coefficients, the HR image x could be found by applying inverse of the contourlet transform [8] to $\widetilde{x}$. Our approach is expressed in block diagram in Fig. 2.1.

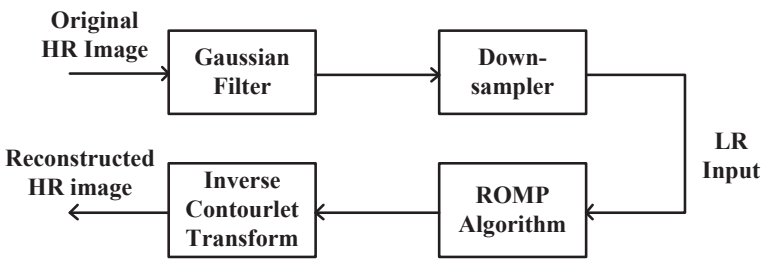


Figure 2.1: Block diagram of proposed approach

As explained earlier in order to have a unique result in the underdetermined equation (2.2), $\widetilde{x}$ is considered to have minimum number of non zero coefficients. Therefore the desired $\widetilde{x}$ is found by solving an l_0-optimization problem. Greedy algorithms introduced in section 2.1, provide a solution for l_0-norm optimization. Therefore the

l_0-optimization problem is expressed as follow:

$$min\|\widetilde{x}\|_0 \quad s.t \quad \widehat{x} = \varphi\widetilde{x} \tag{2.5}$$

We used ROMP as the greedy algorithm for solving equation (2.5). This method is similar to OMP just instead of choosing one coefficient, a number of coefficients are chosen in each iteration. Coefficients are chosen in a way that the group has the largest energy and the largest coefficient in the group can be at most twice larger than the smallest coefficient. The ROMP algorithm is shown in the next page.

As stated in section 2.2 in order to have an acceptable measurement matrix, coherency between A and ψ needs to be reduced. In fact to have a unique solution to an underdetermined system, in addition to have sparsity of the solution vector, it is necessary to have incoherency between matrices A and ψ. There are some known matrices which do meet the RIC condition, like random *Gaussian*, *Bernoulli* and *partial Fourier Matrices*[37].

The sensors in a camera act very similar to a smoothing filter, in fact capturing a scene is the process of low-pass filtering or blurring and then down sampling that scene. Also when the image is down sampled in a computer by software low pass filter is applied prior to down sampling. This blurring effect is very much like applying a Gaussian filter. Considering this observation in the model, not only best describes the hardware characteristics for capturing a video but also increases the incoherency between A and ψ. Consequently in this work a Gaussian smoothing filter (G) is used to represent this characteristic of captured scenes.

Regularized Orthogonal Matching Pursuit (ROMP)

Input: measured vector $\widehat{x}$, level of sparsity k, forward measurement matrix φ, and backward φ^*.

Output: Index set $\mathbf{I} \subset \{1, \ldots, t\}$ which contains non-zero coefficients and high resolution image in transform domain, $\widetilde{x}$.

Initialization: $\mathbf{I} = \{\}$ and residual $r = \widehat{x}$.

While $r < \varepsilon$

• $u = \varphi^* r$ and $J \leftarrow$ set of k largest magnitude coefficients of u sorted in a non-increasing order.

• Among all subsets $J_0 = \{i, \ldots, j\}, \{J(i), ..., J(j)\} \subset J$ where $J(j) \geq \frac{1}{2}J(i)$, Choose the one which gives maximum energy in J from element i to j.

• Add subset J_0 to index set $\mathbf{I}$ ($\mathbf{I} \leftarrow \mathbf{I} \cup J_0$)

• $y = argmin\|\widehat{x} - \varphi z\|_2$ and $supp(z) = \mathbf{I}$

$r = \widehat{x} - \varphi y$

end

To check the effect of this filter on coherency reduction the coherency is computed between A and ψ, with and without Gaussian matrix. The coherency is found by taking the maximum inner product between any two basis elements of the bases multiplied by $\sqrt{n}$ [36]. In this experiment n is equal to 256×256.The coherency without Gaussian matrix is equal to 193.88 while when Gaussian filter is considered, it will drop to 159.48. This reduction can give us the permission to apply CS framework to our problem [10]. Also the Gaussian matrix will cause the HR image to become blurred which is actually the process that happens in the camera when capturing a

scene [10][37]. Also it satisfies the RIC condition for applying sparsity framework. Consequently equation (2.5) would be modified to:

$$min\|\widetilde{x}\|_0 \quad s.t \quad \widehat{x} = AG\psi\widetilde{x} \tag{2.6}$$

As mentioned in the ROMP algorithm we do need both the forward measurement matrix (φ) and backward measurement matrix (φ^*). By having $\varphi = AG\psi$, φ^* can be computed:

$\varphi^* = \psi^T G^* A^T$.

The output of the algorithm ($\widetilde{x}$) is in contourlet domain; by taking inverse of contourlet transform the HR image would be recovered. Our results show improvement over conventional methods for interpolation like bicubic and cubic spline and newer methods introduced in[7], [10] and [11].

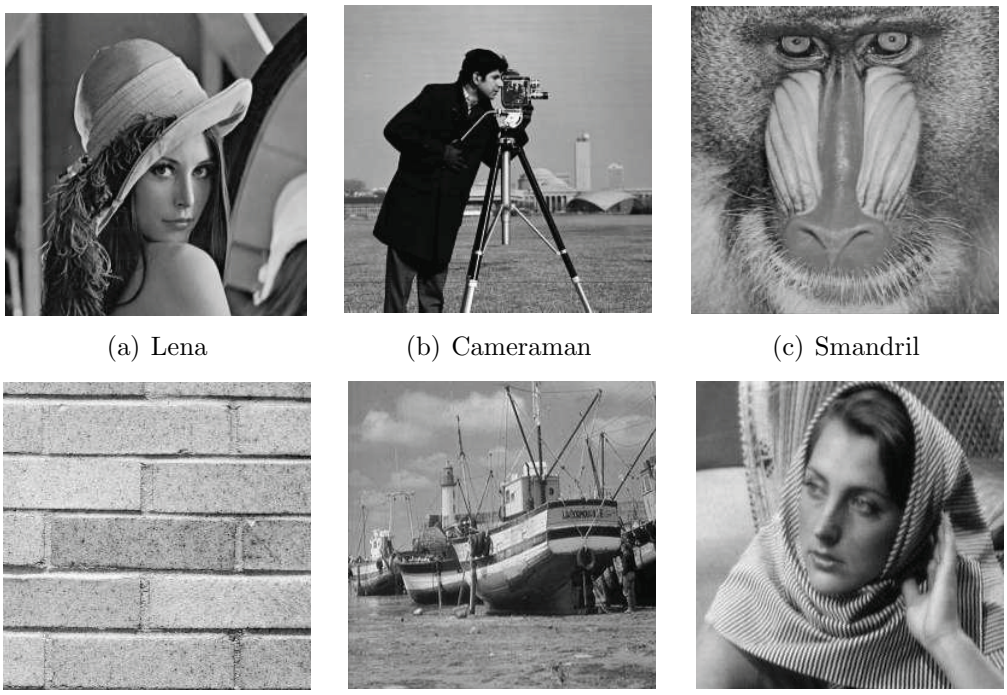
2.4 Experimental Result

Number of levels of contourlet decomposition and number of directions in each level has impact on the quality of the output. The number of levels of decomposition can be at most $log_2(l) - 2$ where the image is of size $l \times l$. Therefore the finest scale would be of size 4×4; but increase in the number of levels of decomposition and the number of directions will cause the program to be slow. Consequently while the quality of the reconstructed images is not degrading, we prefer to keep the number of levels and directions as small as possible. We found experimentally that setting number of levels equal to $\lfloor((log_2(l) - 2)/2)\rfloor$, preserves the speed of the program and also yields the best results both in PSNR and visual quality. Number of directional decomposition

at the first stage is considered to be 1 and then at each pyramidal level this value is doubled (from coarser to finer scale).

The algorithm has been implemented in MATLAB and tested on different images and the PSNR values are given on table 2.1. The images used in table 2.1 are shown in Fig. 2.2. Original HR images are of size 256×256. For constructing the LR input images, HR images were filtered and then down sampled by a factor of 2. Initial estimates of the HR images are constructed by up sampling using zero-filling and then using a Gaussian filter to make the estimated HR image smooth. The standard deviation of the Gaussian filter is proportional to the resolution factor. This estimated HR is the input to ROMP algorithm. Final estimate is computed by applying inverse of the contourlet transform to the output of the ROMP algorithm.

We have compared our algorithm with conventional methods like "bicubic interpolation" and "cubic spline". Also the results are compared to more recent works like edge directed interpolation of [7] and the wavelet based work of [10] and the work done by Mallat and Yu in [11]. In Fig. 2.3 the result of different methods on *barbara* image is shown and the final outputs for the *wall* image can be seen in Fig. 2.4. It can be seen that our results contain sharper edges and more details and the outputs are more natural while in other methods they are blurred or not clear. As it can be seen from the values in table 2.1 our results outperform all other approaches in terms of PSNR. On average, our method is 3.05dB better than bicubic interpolation and 1.39dB better than cubic spline. In comparison to newer methods our approach is 2.36dB better than method in [7], 1.82dB better than method in [10] and 0.93dB better then [11].



(a) Lena (b) Cameraman (c) Smandril

(d) Wall (e) Boat (f) Barbara

Figure 2.2: Sample Images.

Table 2.1: PSNR of recovered HR images by different algorithms.

Methods	Lena	Cameraman	Smandril	Wall	boat	barbara	Average
Bicubic	26.47dB	24.25dB	25.31dB	24.93dB	25.58dB	23.92dB	25.08dB
Cubic Spline	29.16dB	25.28dB	26.06dB	26.80dB	27.67dB	25.44dB	26.74dB
Nedi [7]	27.45dB	24.83dB	25.58dB	26.11dB	26.44dB	24.21dB	25.77dB
Wavelet-based [10]	28.00dB	25.51dB	25.89dB	27.03dB	27.16dB	24.25dB	26.31dB
SME [11]	30.07dB	26.23dB	26.50dB	27.02dB	28.34dB	25.06dB	27.20dB
Our approach	**30.19dB**	**26.89dB**	**27.71dB**	**28.56dB**	**29.02dB**	**26.41dB**	**28.13dB**

(a) Original HR

(b) Bicubic

(c) Cubic Spline

(d) Method in[7]

(e) Method in [10]

(f) Method in [11]

(g) Our Approach

Figure 2.3: Results of different approaches for Super Resolution on *Barbara* image.

(a) Original HR

(b) Bicubic

(c) Cubic Spline

(d) Method in[7]

(e) Method in [10]

(f) Method in [11]

(g) Our Approach

Figure 2.4: Results of different approaches for Super Resolution on *Wall* image.

Chapter 3

Video Resolution Enhancement

In this chapter we have extended our work on image interpolation to the video super resolution. The method introduced in the previous chapter can be used on single video frames to make a HR frame out of a LR frame. But in a video sequence since there is correlation between the adjacent frames this correlation can be used as an additional knowledge of the original high resolution frame and will help to find a HR frame with better quality and fewer artifacts.

Therefore we first applied our image super resolution method to each frame of the sequence and then propose a post processing step to improve the output both is subjective and objective quality. Proposed approach is shown in a block diagram in Fig. 3.1.

The post processing step is a 3-D Bilateral Total variation filter which is an extended version of the 2-D BTV filter introduced in [1].

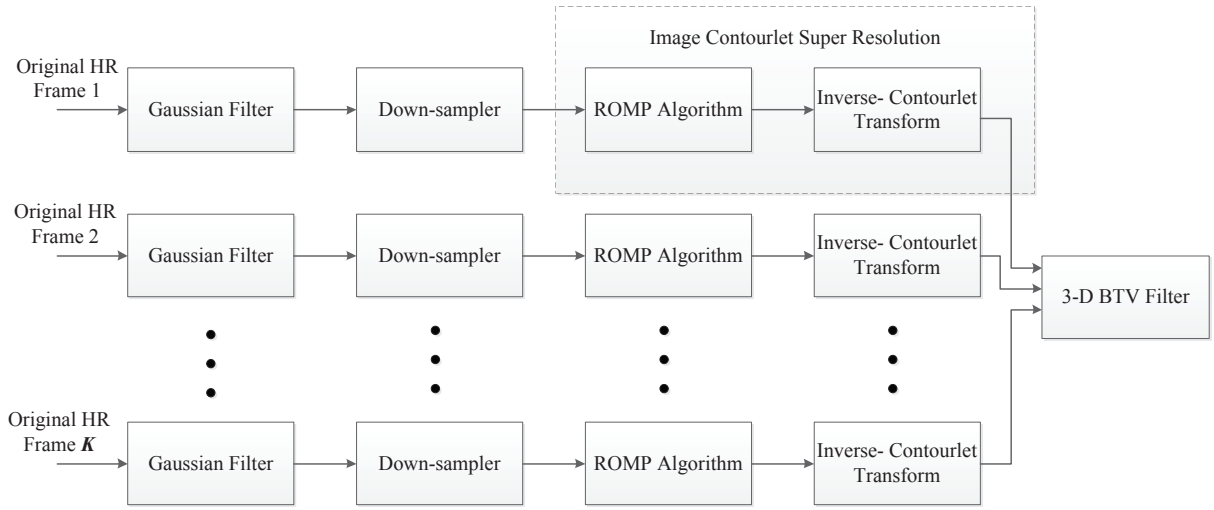


Figure 3.1: Block diagram of Video Resolution Enhancement Method

3.1 Deblurring and Regularization Term

Because the captured scene has movement and this movement may be visible even on each frame depending on the aperture size of the camera and the acquisition time the reconstructed frame might be blurred. Therefore our post processing step contains a deblurring term and also a regularizer term. The function for deblurring and regularization is as follow:

$$X = argmin[\|GX - Z\|_1 + \lambda R_{BTV}(X)] \tag{3.1}$$

where X is the desired high resolution video sequence, Z is the current interpolated video sequence and G applies Gaussian blur matrix on each frame of the sequence. In the second part of the statement, λ is the regularizer parameter and $R_{BTV}(X)$

is the regularization function which is the 3-D extension of Bilateral Total Variation introduced in [1].

$$R_{BTV}(X) = \sum_{l=-P}^{P} \sum_{m=-P}^{P} \sum_{k=-P}^{P} \alpha_1^{|m|+|l|} \alpha_2^{|k|} \times \|(X - S_x^l S_y^m S_t^k X)\|_1 \tag{3.2}$$

In equation (3.2) P is the window size which can be a different value for spatial domain and temporal domain and S_x^l, S_y^m and S_t^k, shift the sequence X, l, m, k pixels in horizontal, vertical and temporal directions respectively. Also $\alpha_i (i = 0, 1, 0 < \alpha_i < 1)$ gives a decaying effect to the summation of the regularization term.

Thus the problem would be to solve equation (3.2).

$$X = argmin[\|GX - Z\|_1 + \lambda\{\sum_{l=-P}^{P} \sum_{m=-P}^{P} \sum_{k=-P}^{P} \alpha_1^{|m|+|l|} \alpha_2^{|k|} \times \|(X - S_x^l S_y^m S_t^k X)\|_1\}] \tag{3.3}$$

For finding minimum of the *argmin* the gradient of equation (3.3) is computed. By using steepest descent method the solution to this problem would be:

$$\begin{aligned} X_{n+1} &= X_n - \beta\Big\{G^T sign(GX_n - X_{int}) + \\ &\lambda \sum_{l=-P}^{P} \sum_{m=-P}^{P} \sum_{k=-P}^{P} \alpha_1^{|m|+|l|} \alpha_2^{|k|} (I - S_t^{-k} S_y^{-m} S_x^{-l}) \times sign(X_n - S_x^l S_y^m S_t^k X_n)\Big\} \end{aligned} \tag{3.4}$$

where X_{int} is the initial input which is the output of the single frame interpolation step and X_n is the current estimate of the HR frame, where n is the iteration number. β is the step size in the direction of the gradient.

In generalizing the 2-D BTV to 3-D version different decaying parameter have been allocated to the spatial and temporal shifts. This is due to the fact that the behavior of a video sequence in spatial (x, y) and temporal (t) domain are different. A

video sequence with fast motion may lose the consistency in time faster than spatial consistency, because frames have motion in respect to each other and thus a single pixel may be more correlated to its adjacent pixels in space than its adjacent pixels in time. Consequently $\alpha_1 > \alpha_2$ seems to give better results in these sequences. But in a sequence where motion is very small and frames do not change much with respect to each other, a single pixel is more correlated to the adjacent frames in time and thus in these cases $\alpha_1 < \alpha_2$ is chosen. Testing different values for α_1 and α_2 supports the specified criteria for choosing α_1 and α_2. Experimental results are driven by choosing the best values for $\alpha_i (i = 0, 1)$ with considering the stated condition.

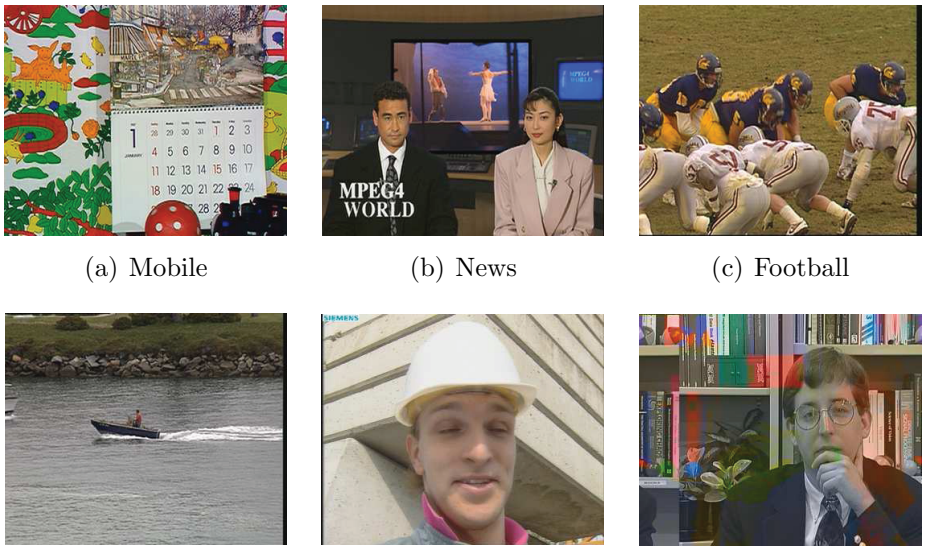
3.2 Experimental Result

In this chapter for video resolution enhancement single frame super resolution is applied on each frame which is same as image super resolution method proposed in chapter 2. For constructing the LR video sequence, HR frames are filtered and then down sampled by a factor of 2. Each LR frame constructed in this way then goes through our image super resolution algorithm and the output is the reconstructed HR video sequence. Then the 3-D Bilateral Total Variational filter along with deblurring term is applied to the output sequence. In 3-D filter the criteria for parameter selection is to choose parameters which produce visually most appealing result therefore to ensure fairness each experiment has been repeated with different parameters (α, β, λ) to find the best result. Number of iterations for the post processing step is equal to 70 and P in equation (3.4) is equal to 2.

The proposed algorithm has been compared with lanczos interpolation, NEDI algorithm [7] and DCT-based method [9]. Also the results are compared to more recent

works like the MASK method done by Takeda, Van Beek [17] and the hybrid DCT-Wiener based method [20]. Table 3.1 is the PSNR comparison of different methods on 4 CIF video sequences. The video sequences used in this experiment are "*mobile*", "*news*", "*football*" and "*coastguard*". In Fig. 3.2 the first frames of the video sequences used for comparison are shown. Also the PSNR values of each frame are given in a separate graph in Fig. 3.3.

Another comparison has been made with the real-time DSP-based method [21]. Since the code for this method was not available a separate comparison has been made and the same video sequences used on that paper are used. The videos are "*foreman*", "*mobile*" and "*deadline*" and are in QCIF format. Table 3.2 is the comparison of the proposed method and the real-time DSP-based method [21].

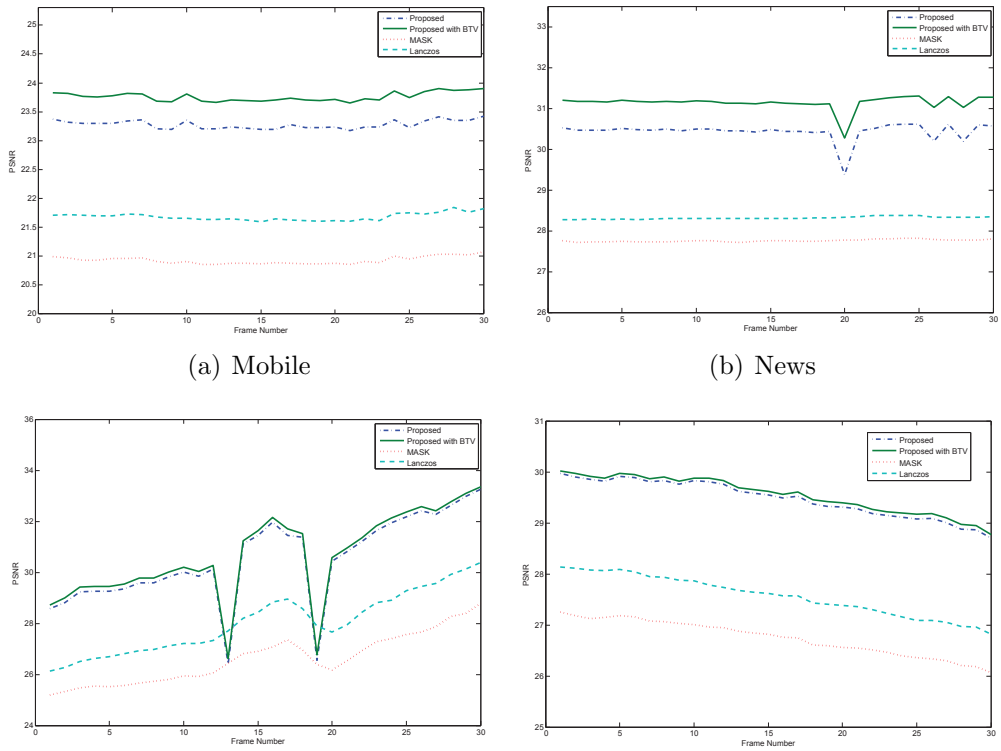


(a) Mobile (b) News (c) Football

(d) Coastguard (e) Foreman (f) Deadline

Figure 3.2: Video sequences used for comparison

The subjective quality of the proposed approach is compared to other methods in Fig. 3.4 and Fig. 3.5. Also to clarify the quality a part of the "*news*" is shown in



(a) Mobile (b) News

(c) Football (d) Coastguard

Figure 3.3: PSNR values for each frame in the used video sequences

Fig. 3.6.

Table 3.1: Average PSNR comparison for different algorithms using CIF video sequences

Methods	Mobile	News	Football	Coastguard
Lanczos Interpolation	21.70	28.31	28.03	25.09
Nedi [7]	20.34	26.07	25.45	25.58
DCT [9]	22.74	29.8	29.61	28.6
MASK [17]	20.93	27.76	26.62	26.74
Hybrid method in [20]	23.32	30.65	30.76	29.25
Proposed approach	**23.76**	**31.14**	**30.85**	**29.47**

Table 3.2: Average PSNR using QCIF video sequence

Sequence	Method in [21]	Proposed approach
Foreman	31.00	31.87
Mobile	23.07	23.54
deadline	23.32	24.72

(a) Lanczos

(b) MASK

(c) Proposed Method

(d) Original HR

Figure 3.4: Results of different approaches on 10-th frame of "*news*" sequence

(a) Lanczos

(b) MASK

(c) Proposed Method

(d) Original HR

Figure 3.5: Results of different approaches on 15-th frame of "*mobile*" sequence

(a) Lanczos

(b) MASK

(c) Proposed Method

(d) Original HR

Figure 3.6: Cropped 10-th frame of "*news*" sequence

Chapter 4

Deinterlacing

Interlaced video consists of a sequence of fields instead of frames. Odd fields are formed from odd lines and even fields are formed from even lines. In the deinterlacing process our goal is to convert these fields into frames. In chapter 1 a number of deinterlacing methods were reviewed.

In this chapter for video deinterlacing problem we have proposed a simple approach in combination with Bilateral Total Variation Filter (introduced in the previous chapter). Images are generally smooth and neighboring pixels are highly correlated. We have used this correlation for the video deinterlacing problem. We have considered an available line as a representative for the below unavailable line. Assuming that they are highly correlated this can be very helpful in estimating unknown lines in deinterlacing. However since each part of a line may belong to certain object in the image and each line passes through different parts of different objects in the image, considering a set of two lines (available and unavailable) as a correlated pair might not be completely correct. Instead we consider a window on an unavailable line and assume that the pixels in the window are highly correlated to the pixels in the same

window but in the above line which is an available line. This can be a more reasonable assumption.

Our proposed deinterlacing approach consists of two steps. In the first step each frame of the sequence is deinterlaced using a number of simple interpolators. We considered a sliding window that it's size will be discussed later. This window covers the pixels in an unavailable line that we are interpolating at each instance. We call this window the unknown window. Moving this window to the above line (which is an available line) the new window would be called representative. In fact since all the values in the representative window are available we use this window as a reference. So each time we want to measure goodness of an interpolator for the unknown window we compute the difference between interpolated pixels obtained from that interpolation method and the representative window. We choose the one with minimum error among these interpolators. In the next step which is a post processing step we applied Bilateral Total Variation filter introduced in previous chapter. In this chapter we have modified BTV filter to match the deinterlacing problem. The details of these two steps are given in the next two sections.

4.1 First Step

We have used 6 different interpolators which are shown in Fig. 4.1. Each interpolator is the mean value of two pixels shown by the same number in Fig. 4.1. Three of these interpolators (numbers 1,3,4) are spatial and the other three(2,5,6) are temporal.

All the pixels in an unknown window in a frame of an interlaced video sequence is reconstructed using all of these interpolators. Then for a specific window size the *sum of absolute difference* (SAD) between the representative window and the interpolated

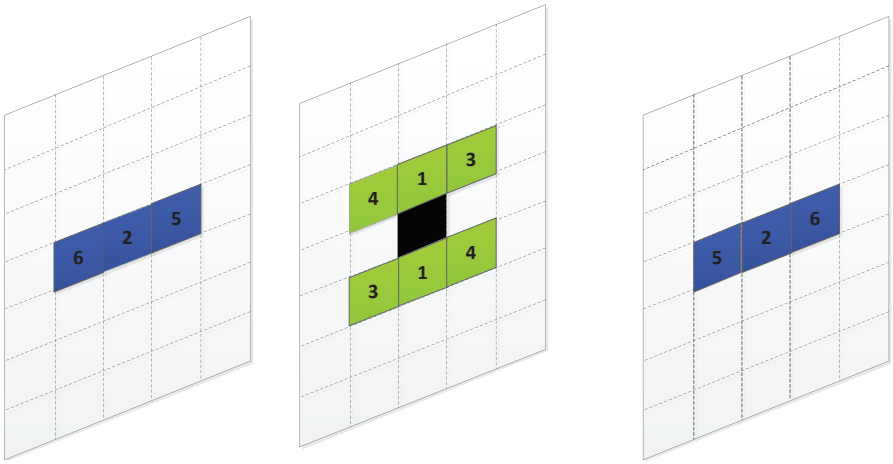


Figure 4.1: 6 interpolators used in the deinterlacing algorithm

section is calculated. Equation 4.1 formulates SAD computation for each spatial interpolator(interpolators 1,3,4).

$$SAD(i,j,n) = \Sigma(|V(i-1, j:j+k, n) - Xs(i, j:j+k, n)|) \tag{4.1}$$
$$i = 3,5,7,...(if\ \ n = 2p) \quad i = 2,4,6,...(if\ \ n = 2p+1)$$

Xs is the deinterlaced video using one of spatial methods and V is the interlaced video with zeros on the lines which are not available and k is the size of the window. SAD is computed for k columns of each available line $(i-1)$ of each field (n). In equation (4.1), j indicates the starting pixel of the window.

For the temporal interpolators SAD is computed between the representative and

the computed line segment in the previous field. This is due to the fact that the representative window is actually a representative of the unknown window.

$$SAD(i,j,n) = \Sigma(|V(i-1, j:j+k, n) - Xt(i-1, j:j+k, n-1)|) \tag{4.2}$$
$$i = 3,5,7,...(if\ n = 2p) \quad i = 2,4,6,...(if\ n = 2p+1)$$

In equation (4.2) the formulation of the SAD for temporal methods is given in which Xt is the deinterlaced video using one of the three temporal interpolators. So in temporal methods we do not work with the window for which we want to choose the best interpolator as a term in the SAD computation. Instead we use the representative window for SAD calculation. Consequently in equation (4.2) we are computing the difference between $V(i-1, j:j+k, n)$ (The representative window) and $Xt(i-1, j:j+k, n-1)$ which is the temporal deinterlaced window in the previous field and in the same horizontal position as the representative. Fig. 4.2 shows the pixels which are used for SAD computation in each interpolator.

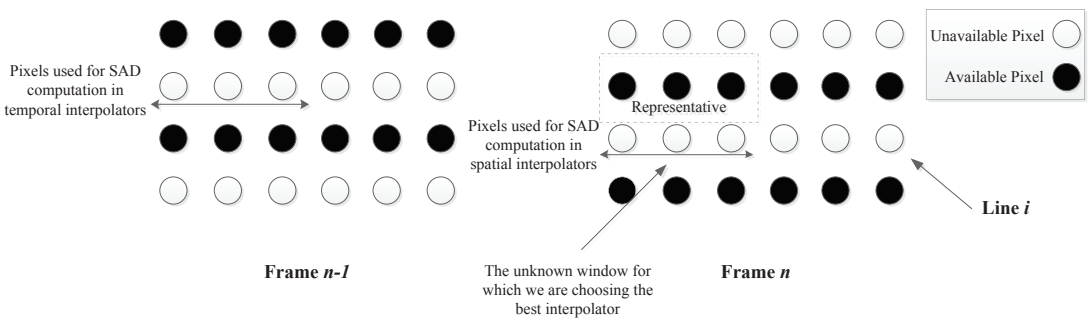


Figure 4.2: SAD computation in proposed method

Choosing the value of k (size of the window) depends on the characteristics of the video. Whether the video has significant correlation between adjacent pixels affects the choice of k. We have proposed an approach so that the algorithm chooses the value of k between 3 predetermined sizes ($k = 8, 16, 32$) adaptively. In order to force the proposed method to adaptively select value k, we have set $k = 32$ as an initial value. Then it computes SAD for each interpolator for this special window size and divide it by the number of pixels, this will give us the average error for each pixel. This parameter is then computed for all interpolators and for $k = 16$ and $k = 8$. The best window size and the best interpolator is the one which gives minimum average error per pixel among all different cases.

It is clear that for each line, there might be different window sizes so there is not a fixed value k for each frame or even for each line. Also it should be noted that the last window of each line is chosen not based on the average error per pixel but based on the pixels left from the entire row.

4.2 Post Processing

In this step we have applied a post processing to further improve the quality of deinterlaced video. The proposed method is based on the Bilateral Total Variation (BTV) filter introduced in the previous chapter. We have modified BTV filter to match the problem of deinterlacing.

Equation (4.3) is the formulation of the proposed post processing step. The post processing term is a regularizer which is applied on each single frame of the deinterlaced video.

In this term we only modify the pixel values of the lines which were not available in

the original interlaced frame and we do not change the available lines.

$$\begin{aligned} X_{q+1}(i,:,n) &= X_q(i,:,n) - \\ &\delta\Big\{\Sigma_{m=-P}^{P}\Sigma_{l=-P}^{P}\alpha_i^{|m|+|l|}[F_1(X_q(i,:,n),m,l) - F_2(X_q(i,:,n),m,l)]\Big\} \end{aligned} \tag{4.3}$$

where :

$F_1(X,m,l) = sign(X - S_x^l S_y^m X)$

$F_2(X,m,l) = S_y^{-m} S_x^{-l}(sign(X - S_x^l S_y^m X))$

The index q in X_q refers to the iteration number. i in above equation is an even value if X_q is an odd frame and vice versa. δ is the regularization parameter and $X_q(i,:,n)$ refers to i-th line of the frame n in q-th iteration. α_i is a scaler to give decaying effect to the summation but here we have assigned a different value of α to each line. S_x^l and S_y^m are the shift operators which shift X by l and m pixels in horizontal and vertical direction. In equation (4.3) in first iteration X_1 is equal to the deinterlaced frame from the previous step.

α_i is set to be proportional to the sum of differences between the corresponding line and the known line above it. Consequently if the difference between the reconstructed line and the above line is not very large, α_i would be small and the interpolated values will not change a lot in the post processing step. But if the difference is a large value α_i would be a large value. We always set α_i 's which are larger than 1 equal to a predetermined value that depends on the video.

4.3 Experimental Result

In this section we compare our proposed deinterlacing algorithm with other algorithms in literature both objectively and subjectively. Table 4.1 is the PSNR comparison of the proposed method with and without 2-D BTV filter with methods already available in literature. The first frame of the videos used for PSNR comparison are shown in Fig. 4.3.

(a) Foreman (b) Mother (c) Mobile (d) Hall

(e) Stefan (f) Silent (g) Container (h) Miss

Figure 4.3: Video sequences used for deinterlacing

In this experiment for the post processing step number of iterations is equal to 70 and P in equation (4.3) is equal to 2. Also δ is chosen in a way to give the best result. Therefore the algorithm has been tested for different values of δ and the most visually appealing result was chosen.

The proposed algorithm has been compared with simple methods like line averaging and edge-based line averaging (ELA). Also it has been compared to more advanced methods such as DIMC method proposed in [39], MOMA method [40] and 4FLMC method proposed in [42].

Table 4.1: Average PSNR comparison for different deinterlacing algorithms using 8 CIF videos

Methods	Foreman	Mother	Mobile	Hall	Stefan	Silent	Container	Miss
LA	32.34	39.22	25.45	31.74	27.46	33.79	28.75	43.59
ELA [38]	31.80	39.02	23.58	30.77	25.96	33.02	27.75	42.75
DIMC [39]	33.09	39.91	24.65	31.70	27.26	36.52	29.15	40.85
MOMA [40]	32.93	42.62	28.21	39.04	26.53	38.16	44.22	40.64
MCAMA [41]	34.06	43.26	25.91	38.08	27.16	40.31	40.09	NC
4FLMC [42]	34.65	45.38	27.40	40.62	27.18	40.91	41.96	41.53
HMA [43]	34.83	42.56	28.42	38.33	27.06	39.02	NC	NC
NNDMF [44]	34.26	42.57	**28.93**	36.99	26.18	37.36	39.61	39.63
TMCD [45]	33.77	44.95	27.66	NC	**28.79**	NC	45.49	NC
Proposed	35.65	46.19	28.33	40.65	28.13	41.69	47.09	44.54
Proposed with BTV	**35.66**	**46.25**	28.33	**40.77**	28.14	**41.70**	**47.11**	**45.25**

The results of some newer methods are also given in table 4.1 like MCAMA method proposed by H. Sun, et al [41], HMA method [43], NNDMF method proposed in [44] and the TMCD method introduced in [45]. Since the codes for these methods were not available we had to use the values given in their papers and the PSNR values for some videos were not available; consequently we put *NC* (Not Considered) for those videos. As you can see in the table the proposed approach outperforms other methods for most videos and for the videos for which PSNR of the proposed method in not on top, it is not that far.

For visual comparison the proposed approach is compared to some of the methods named in table 4.1. We have chosen the methods for which the codes were available or we have written the codes. Result of the proposed deinterlacing methods and some other methods for 28-th frame of the *mother* sequence is shown in Fig. 4.4. It can be seen in Fig. 4.4 that the result for *LA* and *ELA* method is somehow blurred and for

the *4FLMC* method there are some artifacts specially near the woman's mouth. But in our proposed approach there is no artifact and the quality is much better than LA or ELA. For more comparison we have shown the results for 100-th frame of *foreman* sequence on Fig. 4.5. In order to make the comparison more clear a part of the *foreman* frame on Fig. 4.5 is zoomed in and is shown in Fig. 4.6. In this figure the ringing effect in LA and ELA frame and also the artifacts on the mouth of the man is more clear.

(a) LA

(b) ELA

(c) 4FLMC

(d) Proposed method with BTV

(e) Original Frame

Figure 4.4: Result of deinterlacing on 28-th frame of *mother* sequence

(a) LA

(b) ELA

(c) 4FLMC

(d) Proposed method with BTV

(e) Original Frame

Figure 4.5: Result of deinterlacing on 100-th frame of *Foreman* sequence

(a) LA

(b) ELA

(c) 4FLMC

(d) Proposed method with BTV

(e) Original Frame

Figure 4.6: Zoomed part of *foreman* figure

Chapter 5

Conclusion

In this thesis we worked on the problems of image and video super resolution and also video deinterlacing. These problems are open problems in the field of video and signal processing. Many methods have been introduced in this area but still there is room for improvement.

The problem in the mathematical language is to find a number of unknowns which are much more than our known data. In the image super resolution for example number of unknown pixels is r^2 times number of known pixels, where r is the resolution improvement factor.

In single frame image super resolution the number of unknowns is much more than the number of known pixels. This underdetermined problem has infinite answers, in order to restrict the output to a single solution we need to add more constraints. Compressive sensing offers a solution to the underdetermined problems which are in a sparse domain. In fact if the signal is sparse under some basis, CS can help to find a unique solution for this problem by adding a constraint. The constraint that CS considers for the output of underdetermined problems is to have minimum number of

coefficients in that sparse domain. We used this paradigm in chapter 2 to introduce our image interpolation approach.

To transform the image to a sparse domain in order to be able to use compressive sensing, we have used contourlet transform. In this sparse domain we find the solution to the underdetermined problem of super resolution by using CS and finally by taking inverse of contourlet transform we found the super resolved image.

As shown in chapter 2 the result of the proposed image super resolution method outperforms other existing methods both in subjective and objective quality. Also since the contourlet transform preserves edges an advantage of the proposed method is that it contains sharp edges.

In the next chapter we generalized proposed image super resolution method to video resolution enhancement. For this purpose we applied the introduced image super resolution method on each frame separately and then we applied a 3-D Bilateral Total Variation filter on the sequence. We have generalized the 2-D BTV filter introduced in [1] as a regularization term to a 3-D filter which is applied on x, y and t directions. This filter helps to have better results than the single frame super resolution since it considers the variations to be small in both spatial and temporal domains.

The results for the video resolution enhancement method are compared to some other methods in chapter 3. The results are better than those methods both visually and in PSNR value.

In chapter 4 we proposed a deinterlacing approach. The proposed approach consists of 2 steps. The first step is based on some simple interpolators. In this step we tried to choose the interpolator which best interpolates the missing lines. In fact among all 6 interpolators introduced in chapter 4 the one which gives the minimum

error is chosen. Then the deinterlaced frame computed in this step is fed to the post processing step which is a modified version of 2-D BTV filter.

The computed output is then compared to several methods currently available in literature, both motion compensated and non motion compensated methods. The table of PSNR values is given, also the frames can be visually compared by the given figures. As results show our proposed approach outperforms other methods in most cases both subjectively and objectively.

References

[1] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. on Image Processing,* vol. 13, no. 10, pp. 1327-1344, Oct. 2004.

[2] A. R. Pais, J. D'Souza, G. R. M. Reddy and S. B. Patil,"Super Resolution of Video with sharpened edges using Multiple Frames - A Novel Approach", International Conference on Advances in Recent Technologies in Communication and Computing, pp.98-102,2009.

[3] H. Stark and P. Oskoui, "High resolution image recovery from imageplane arrays, using convex projections," *Journal of Optical Society of America A,* vol. 6, no. 11, pp. 1715-1726, 1989.

[4] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Processing,* vol. 53, no. 3, pp. 231-239, 1991.

[5] C. V. Jiji and S. Chaudhuri, "Single-frame image super-resolution through contourlet learning," *EURASIP J. Appl. Signal Processing,* pp.235-235, 2006.

[6] X. Zhang, X. Wu, "Image interpolation by adaptive 2D autoregressive modeling

and soft-decision estimation," *IEEE Trans. on Image Processing,* vol. 17, no. 6, pp. 887-896, 2008.

[7] X. Li and M.T. Orchard, "New Edge Directed Interpolation," *IEEE Trans. on Signal Processing,* vol. 10, no. 10, pp.1521-1527, 2001.

[8] L. Zhang and X. Wu , "Image interpolation via directional filtering and data fusion,"*IEEE Trans. on Signal Processing,*vol. 15, no. 8, pp.2226-2238, 2006.

[9] H. W. Park, Y. S. Park and S. K. Oh, "L/M-fold image resizing in block-DCT domain using symmetric convolution," *IEEE Trans. on Image Processing,* vol. 12, no. 9, pp. 10161034, 2003.

[10] P. Sen and S. Darabi, "Compressive Image Super Resolution," *IEEE Asilomar,* pp.1235-1242, 2009.

[11] S. Mallat and G. Yu, "Super-resolution with sparse mixing estimators, *IEEE Trans. on Image Processing,* vol.19, no.11, pp.2889-2900, 2010.

[12] S. Yang, Y, Kim and J, Jeong, "Fine Edge-Preserving Technique for Display Devices,"*IEEE Trans. on consumer electronics,* vol.54, no.4, 2008.

[13] N. Mueller, Y. Lu, and M. N. Do, "Image interpolation using multiscale geometric representations,"*Proc. of SPIE Conference on Electronic Imaging,* 2007.

[14] M.Elad and A.Feuer, "Restoration of single super-resolution image from several blurred, noisy and down-sampled measured images," IEEE Trans. on Image Processing, vol. 6, pp. 1646-1658, Dec. 1997.

[15] Y.Ma, H.Zhang and Y.Xue , "A novel Super-Resolution image reconstruction based on MRF," IEEE CISP, pp.1-4,Oct 2009.

[16] H.Takeda, S.Farsiu and P.Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Processing,*vol. 16, no. 2, pp.349-366, 2007.

[17] H. Takeda, P. van Beek, "Spatio-temporal video interpolation and denoising using motion-assisted steering kernel (MASK regression," *proceeding of International Conf. on Image Processing,* pp.637-640, 2008.

[18] X.Zhang, R.Xiong, S.Ma and W.Gao, "A Robust Video Super-Resolution Algorithm," *Picture Coding Symposium,* 2010.

[19] H. Takeda, P. Milanfar, M. Protter, and M. Elad, "Super-Resolution without Explicit Subpixel Motion Estimation," *IEEE Trans. on Image Processing,* vol. 18, no. 9, pp.1958-1975, 2009.

[20] Z.Wu, H.Yu and C.W.Chen, "A New Hybrid DCT-Wiener-Based Interpolation Scheme for Video Intra Frame Up-Sampling," *IEEE Signal Processing Letters,* vol. 17, no. 10, 2010.

[21] S. Lopez, G. M. Callico, F. Tobajas, J. F. Lopez, and R. Sarmiento, "A novel real-time DSP-based video super-resolution system," *IEEE Trans. on Consumer Electronics,* vol. 55, no. 4, pp. 2264-2270, 2009.

[22] G. M. Callico, S. Lopez, O. Sosa, J. F. Lopez, and R. Sarmiento, "Analysis of fast block matching motion estimation algorithms for video super-resolution systems," *IEEE Trans. on Consumer Electronics,* vol. 54, no. 3, 2008.

[23] S. Keller, F. Lauze, and M. Nielsen, "Deinterlacing Using Variational Methods," IEEE Trans. Image Processing, vol. 17, no. 11, pp. 2015-2028, 2008.

[24] S. J. Park, H. Kim, and J. Jeong, "An Efficient Deinterlacing Algorithm Using Wiener Filter and Geometric Duality," IEEE International Symposium on Consumer Electronics, 2009.

[25] M. Li and T. Nguyen, "A de-interlacing algorithm using markov random field model," IEEE Trans. Image Process., vol. 16, pp. 2633-2648, 2007.

[26] E. Bellers and G. de Haan, "De-Interlacing :A Key Technology for Scan Rate Conversion" Amsterdam, The Netherlands: Elsevier, 2000.

[27] 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 visualcharacteristics," IEEE Trans. Consumer Electronics, vol. 46, pp. 999-1006, 2000.

[28] G. Lee, H. T. Li, M.-J. Wang, and H.-Y. Lin, "Motion adaptive Deinterlacing via edge pattern recognition," presented at the Int. Symposium of Circuits and Systems, 2007.

[29] S. D. Baker, S. S. B. Kang, "An MRF-Based Deinterlacing Algorithm with Exemplar-Based Refinement", IEEE Trans. Image Process, VOL. 18, NO.5, 2009.

[30] H. J Cho, Y. S. Lee, S. H. Oh, and S. J. Oh, "A Voting-based Intra Deinterlacing Method for Directional Error Correction", IEEE Trans. Consumer Electronics, Vol. 56, No. 3, 2010.

[31] M.N. Do, "Directional Multiresolution Image Representations," Ph.D. dissertation, Swiss Federal Institute of Technology, Lausanne, Switzerland, 2001.

[32] M.N. Do and M. Vetterli, "The Contourlet Transform: an Efficient Directional Multiresolution Image Representation," *IEEE Trans. on Image Processing,* 14-12, pp.2091-2106, 2005.

[33] P.J. Burt and E.H. Adelson, "The Laplacian Pyramid as a Compact Image Code," IEEE Trans. on Communication, COM-31:532-540, 1983.

[34] R.H. Bamberger and M.J.T. Smith, "A filter bank for the directional decomposition of images: Theory and design," IEEE Trans. on Signal Processing, vol. 40, no. 4, pp.882-893, 1992.

[35] S. Ji, Y. Xue, L. Carin, "Bayesian Compressive Sensing," IEEE Trans. on Signal Processing, VOL. 56, NO. 6, 2008.

[36] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. on Information Theory,* vol.47, no.7, pp.2845-2862, 2001.

[37] D. Needell and R. Vershynin, "Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit," Arxiv preprint arXiv:0707.4203, 2007.

[38] G. de Haan and E. B. Bellers, "Deinterlacing: An overview," Proc. IEEE, vol. 86, no. 9, pp. 1839-1857, 1998.

[39] O. Kwon, K. Sohn, and C. Lee, "Deinterlacing using directional interpolation and motion compensation," IEEE Trans. Consumer Electron. vol. 49, no. 1, pp. 198-203, 2003.

[40] S. Yang, Y. Y. Jung, Y. H. Lee, and R. H. Park, "Motion compensation assisted

motion adaptive interlaced-to-progressive conversion," IEEE Trans. Circuits Syst. Video Technol., vol. 14, no. 9, pp. 1138-1148, 2004.

[41] H. Sun, Y. Liu, N. Zheng, and T. Zhang, "Motion compensation aided motion adaptive de-interlacing for real-time video processing," Asilomar 2008.

[42] Y. Chang, S. Lin, C. Chen, and L. Chen, "Video de-interlacing by adaptive 4-field global/local motion compensated approach," IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 12, pp. 1569-1579, 2005.

[43] Q. Huang, D. Zhao, S. Ma, W. Gao, and H. Sun, "Deinterlacing using hierarchical motion analysis," IEEE Trans. Circuits Syst. Video Technol., vol. 20, no. 5, pp. 673 -686, May 2010.

[44] H. Choi and C. Lee, "Neural network deinterlacing using multiple fields and field-MSEs," in Proc. IJCNN, 2007, pp. 869-872. 2007.

[45] Ying-Ru Chen and Shen-Chuan Tai "True Motion-Compensated De-Interlacing Algorithm" IEEE transaction on circuits and systems for video technology, VOL. 19, NO. 10, 2009.