Denoising of Medical Videos through Probabilistic Model

By

Rubab Fatima Bangsah

A thesis submitted to the faculty of Electrical Engineering Department, Military College of Signals, National University of Science and Technology, Rawalpindi in fulfillment of the requirements for the degree of MS in Electrical Engineering.

SUPERVISOR CERTIFICATE

It is to certify that the final copy of the thesis has been evaluated by me, found as per the specified format and error free.

Date: _____ _____

Col. Dr. Imran Tauqir

ABSTRACT

Wavelet based statistical image denoising is vital preprocessing technique in real world imaging. The existing techniques are based on time-frequency domain where the wavelet coefficients need to be independent or jointly Gaussian. In denoising arena there is a need to exploit the temporal dependencies of wavelet coefficients with non-Gaussian nature.

Here we present a denoising strategy based on Hidden Markov Model (HMM) based on Multiresolution Analysis in the framework of Expectation-Maximization algorithm. Proposed algorithm applies denoising technique independently on each frame of the video. It models Non-Gaussian statistics of each wavelet coefficient and captures the statistical dependencies between coefficients.

Denoised frames are restored inversely by processing the wavelet coefficients. Significant results are visualized through objective as well as subjective analysis.

DEDICATION

I dedicate this report to my beloved parents, and my supervisor, Col. Dr. Imran Touqir for their prayers and encouragement.

ACKNOWLEDGMENT

I thank Allah who provided me with strength and caliber to bring this thesis work to its successful completion.

I am deeply obliged to my supervisor, Col. Dr. Imran Tauqir, for his guidance, unwavering support and confidence in me throughout the course of this thesis work. His time and efforts were very valuable. He contributed significantly in the thesis work and also imparted a lot of knowledge to me. I am also grateful to my Guidance and Evaluation Committee,.Dr. Adil Masood, and Lecturer. Maryam Rasool for dedicating their time and making contributions to this thesis work. I offer my regards and blessings to all of those who supported me in any respect during the completion of this work. Alongside that, I thank the university administration, for facilitating the progress of work at different phases, faculty members of MCS for polishing my knowledgebase, my university mates and friends. May Allah bless them all with eternal happiness.

TABLE OF CONTENTS

Chapter 08

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

1. Mean Square Error**MSE**
2. White Gaussian Noise**WGN**
3. Additive White Gaussian Noise**AWGN**
4. Continuous Wavelet Transform**CWT**
5. Discrete Wavelet Transform**DWT**
6. Expectation Maximization**EM**
7. Gaussian Mixture Model**GMM**
8. Hidden Markov Model**HMM**
9. Probability Density Function**Pdf**
10. Hidden Markov Tree**HMT**
11. Probability Mass Function**pmf**
12. Peak Signal to Noise Ratio**PSNR**
13. Maximum a posterior**MAP**
14. Inverse Discrete Wavelet Transform**IDWT**

CHAPTER 1

INTRODUCTION

Video denoising is based on time-frequency data of a video signal. Image denoising is accomplished by various methods: Time-domain, Frequency-domain, and Time-Frequency combination. Spatial domain methods do not account for the temporal correlation between frames [1-3]. From an image Spatial noise is being successfully removed by Spatial filters resulting in achievement of high gain failed in restoring the edges particularly in less noisy areas [4]. Time domain technique considered inter-frame correlation and performed well for motionless videos. [5].

Temporal filters failed to remove the noise and produced fewer blocking artifacts, causing blurring. On the other hand, in case of motioned videos, a temporal filter was not able to give good results in noise removing and delivered fewer blocking artifacts and caused blurring. Hence improved denoising algorithm is need of time, in order to improve the performance of image processing [6,7,8].

C.P. Loizou [10] recommended linear local statistics filter DsFlsmv, followed by nonlinear geometric filter DsFgf4d, and linear homogeneous mask area filter DsFlsminsc. The proposed method, improved class separation between the asymptomatic and symptomatic classes. However, due to average filtering, sharp features and noisy boundaries were left unfiltered.

Methods in image denoising is primarily centered around wavelet transform. A denoising technique based on Double Density Dual Tree Complex Wavelet Transform (DDDT-CWT),[11] YCbCr and YUV space was implemented as multi-directional wavelet transform, where the edges and structural contents were restored. However, degradation in performance was seen significantly in real time scenarios.

Previously [13] used ‘BKMMSEL’ and ‘BKMAPL’ functions on local BKF density, for noise free modelling of 3D discrete complex wavelet coefficients in each sub-band. The tested video sequences were corrupted with different types of noises i.e. AWGN,

Poisson, non-stationary and speckle noise. Noise reduction was enhanced with increased computational expense.

Sugandha Agarwal made a comparison [15] on the productivity of wavelet based thresholding techniques with presence of speckle noise for different wavelet families i.e. Haar, Morlet, Symlet, Daubechies in de-noising medical imaging resonance of brain was proposed. It was found that wavelet transform was proficiently more better in analyzing images at various resolutions but the edges were not restored and caused blurring.

A HMT model with 2D- DWT (Discrete Wavelet Transform) was implemented using HMT model in context with Expectation-Maximization algorithm [16] independently on each video frame. Thus, bringing about fast execution with less computational time, while the improvement needs to be accomplished for enhanced video denoising.

Michele Claus recommended ViDeNN [17] approach was adopted for Video Denoising. Thereby, combining two systems. Implementing first Single Frame Spatial Denoising.Then Temporal Denoising with a window of three frames, in a single feed-forward procedure. However, the limitations of this method was additional computational power.

1.1 Research already carried out

Medical videos i.e. ultrasound, radiology and capsule endoscopy etc are subject to noise attenuation. To address this, despeckling filters were used on ultrasound videos of common carotid artery (CCA). Filters were based on nonlinear filtering (DsFkuwahara), hybrid median filtering (DsFhmedian), linear filtering (DsFlsmv) and speckle reducing anisotropic diffusion (DsFsrad) filtering [12]. Resulting in better visual quality and improved performance in real time.

A new denoising method for medical images e.g. Ultrasound, X-ray and MRI images, was based on Daubechies Complex Wavelet Transform [9]. SDW14 wavelet significantly removed the noise and preserved the details i.e. the local shifts and orientations were preserved with high computational time.

Denoising in CT images was done by another technique, with edge conservation in tetrolet domain (Haar-type wavelet change) [14]. In this a locally adaptive shrinkage rule was applied on high frequency tetrolet coefficients to lessen noise more viably while preserving the edges and geometrical structures .However, the update procedure was slow for large objects.

1.2 Thesis Statement

Here we present a spatial-temporal filtering framework that considerably removes speckle noise from images and videos.

The proposed strategy manages non-Gaussian behavior of wavelet coefficients that are frequently experienced practically and gives proficient outcomes for despeckling of images too. The results displayed that the proposed strategy not only removed noise but retained almost all structural information of every frame.

1.3 Objective

Primary objective of the thesis was to develop a wavelet based examination of 2D signals using HMM. The proposed model captured the non-Gaussian statistics of each wavelet coefficient and exploited inter scale dependences between them. The secondary objective was to apply expectation maximization algorithm in context with probabilistic graphical models to accomplish signals compression mainly for denoising.

1.4 Methodology

The methodology used is mentioned below

- Primary step was development of wavelet domainmodel forgetting initial properties for wavelet transforms. This scheme was further diversified using probabilistic graphical models toexecute second wavelet transform.
- Secondary step was the generation of an algorithm to carryout de-noising on two dimensional video sequences.

1.5 Advantages

Increased efficiency and reduced computational complexity proved to be useful in many perspectives, i.e.

- De-noising of signals
- In signals classification
- In Detecting signals
- Estimating signals
- Compression of signals

1.6 Areas of Application

This technique has following applications in;

- Image Processing for imaging systems, computers and digital cameras.
- Processing Speech Signals for interpreting and processing words spoken.
- Representing electrical signals as sound in Audio Signal Processing .
- Wireless Communications i.e. demodulation, Control Systems equalization.
- Feature Extraction like speech recognition and image understanding.
- In Compression techniques on Videos compression ,Image compression and Audio compression.

1.7 Thesis Outline

This thesis contains following chapters

Chapter 1:

Gives introduction , problem statement, and thesis objective.

Chapter 2:

Consists of preliminary introduction on Wavelet Transform with basic perception of tactics used for processing statistical signals.

Chapter 3:

Explains in detail the probabilistic models .

Chapter 4:

Discusses the Gaussian Models and their different types.

Chapter 5:

This is about the Hidden Markov model framework.

Chapter 6:

Image Statistical Models for Time-Frequency domain that helps in signal denoising .

Chapter 7:

Presents different results and simulation.

Chapter 8:

Contains Conclusion with future work .

CHAPTER 2

Wavelet Transform Analysis

Since 1950's the Fourier transform was the backbone of transform-based image processing but it was much easier to transmit, compress and analyze the two dimensional signals by wavelet transform. Wavelets are one of the most generalized way to analyze and represent multi resolution images. The wavelet transform methods were mathematically developed in 1980s. These methods can decompose signals of finite energy in spatial domain into a set of to analyze them spatial domain.

Wavelet transform is based on wavelets (small waves) as opposed to Fourier transform whose basis functions are sinusoids. Fourier transform that only provide the frequency information of signals. Information based on time and frequency of a signal obtained through wavelet transform, has better local capacity of the time and frequency.

2.1. Continuous Wavelet Transform(CWT)

CWT is like STFT(Short Term Fourier Transform).In both, the signal gets multiply by the function. Unlike STFT, Fourier transform of windowed signal is not taken and variable size window is used for each spectral component. The CWT presents finer time-frequency localization by making time-frequency presentation of signal.

The CWT of signal z(t) is:

$$Z_w(\tau, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} z(t) \psi^*{}_{\tau,s} \left(\frac{t-\tau}{s} \right) dt \qquad (2.1)$$

And

$$\psi_{\tau,s} = \frac{1}{\sqrt{s}} \psi \left(\frac{t-\tau}{s} \right) \qquad (2.2)$$

Where * denotes complex conjugate of $\psi_{\tau,s}(t)$ and z(t) is the signalto be examined, τ is translation factor and **s** is scaling factor .$\psi_{\tau,s}(t)$is mother wavelet and is the transforming function.

Inverse CWT can be applied to get reconstructed signal .

$$z(t) = \frac{1}{M_\psi^2} \int_s \int_\tau Z_w(\tau, s) \frac{1}{s^2} \psi \left(\frac{t-\tau}{s} \right) d\tau \, ds \qquad (2.3)$$

where M_ψ^2 is a constant called admissibility constant that depends on wavelet used.

Morlet wavelet, Mexican hat wavelet and Paul wavelet are some examples of CWT. Wavelets determined in series in CWT, for reconstruction of a signal that has highly redundant information. Comparison of CWT with DWT shows that DWT has reduced computational complexity while giving appropriate information of actual analyzed signal.

2.2 Discrete Wavelet Transform (DWT)

In DWT, wavelets are sampled discretely. The main advantage over Fourier transform lies in the temporal resolution i.e. capturing time- frequency data. The wavelets gets generated from "mother" wavelet using offsets with dilation parameters in correlation with the two parameters ,

$$f(t) = \sum_x \sum_y c_{xy} \Psi_{xy}(t) \qquad (2.4)$$

Where the expansion coefficients are given by

$$c_{xy} = \int f(t) \Psi_{xy}(t) dt \qquad (2.5)$$

and the condition wavelets obey is given as

$$\Psi_{xy}(t) = 2^{\frac{x}{2}} \Psi(2^x t - y) \qquad (2.6)$$

Here Ψ is mother wavelet, x is dilation parameter and y is offset parameter.

2.2.1 Multiresolution analysis

Multiresolution analysis provides a hierarchical framework for image manipulation and analyzation. Multiresolution involves two-dimensional analysis and decomposition of image data. It was basically designed to work where low frequency elements are more persistent. High frequency elements exist for short durations within a signal. Wavelet analysis is an example of multiresolution analysis.

The signalX[n] is passed via sequence of filters. G_o is low pass filter andH_o is high pass filter, n is an integer. For each level, the high pass filters provide detailed information of input signal X[n]; shown by d[n], whereas low pass filter manages scaling function that gives estimation of the signal a[n].

The filtering process performs the down sampling and continues until the desired level is achieved. The number of levels are determined by the length of signal.

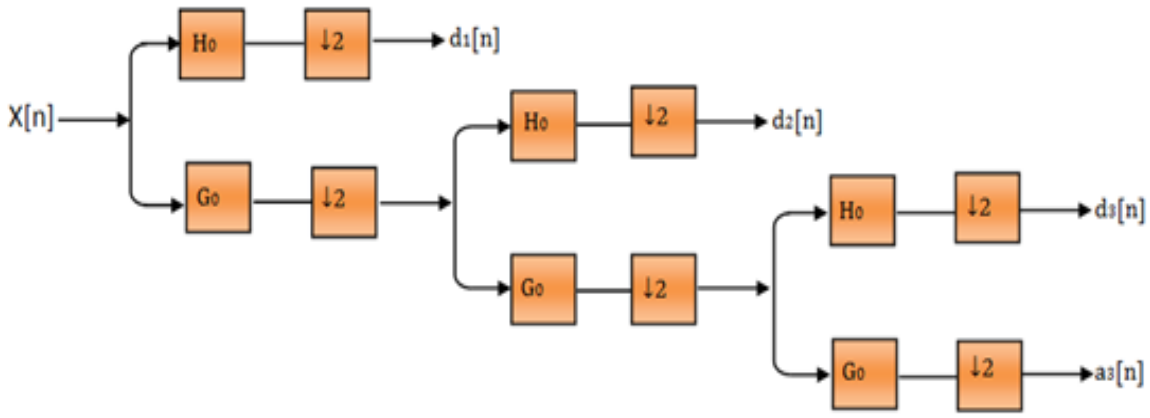


Fig 2.1 Wavelet Decomposition

Reconstruction is accompanied through the upsampling by first passing sampled signal through high pass filters than low pass filters .Then adding simultaneously to get the original or reconstructed signal. G_1 and H_1as synthesis filters.

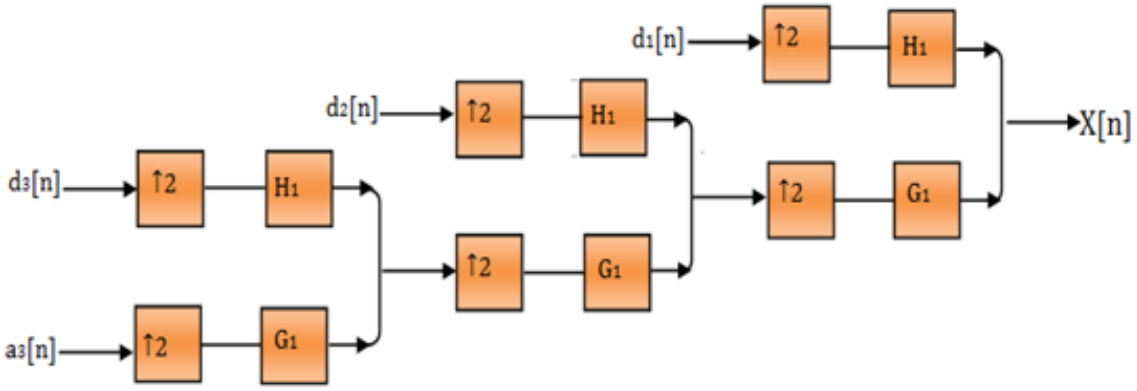


Fig. 2.2 Wavelet Reconstruction

2.2.2 Perfect Reconstruction Conditions

To achieve better remodeling construction ,we need synthesis and analysis of filters for satisfaction of some conditions. $G_o(z)$is low pass analysis filters and $G_1(z)$ as low pass synthesis filters. $H_o(z)$ and $H_1(z)$are high pass analysis and synthesis filters respectively. Following conditions must be fulfilled for complete reconstruction.

$$G_o(-z)G_1(z) + H_o(-z)H_1(z) = 0 \qquad (2.7)$$

$$G_o(z)G_1(z) + H_o(z)H_1(z) = 2z^{-d} \qquad (2.8)$$

The accuracy of perfect reconstruction can be checked through different parameters like Peak Signal to Noise Ratio . In some applications there is no need for reconstruction likely in pattern recognition . However, these applications are not applicable in above mentioned conditions.

2.3 Wavelet Families

Wavelet basis functions often behaves as parent wavelet .The key of having efficiency in wavelet transform is dependent upon choosing right type of mother wavelet. A significant amount of contribution has been put forth by Daubechies, in this work Daubechies 8 wavelet has been used to perform DWT.

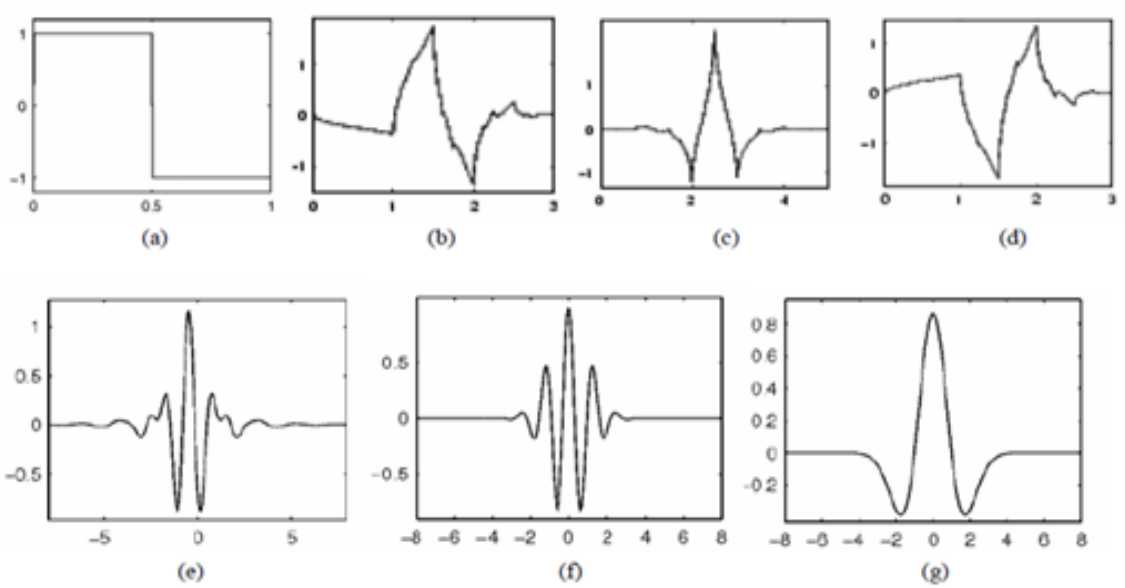


Figure 2.3 Wavelet Families (a) Haar (b) Daubechies (c) Coiflet1 (d) Symlet2 (e) Meyer (f) Morlet (g) Mexican Hat

CHAPTER 3

Probabilistic Graphical Model

Probabilistic graphical models are used to combine probabilities and independence constraints of complex real world scenarios into a compact graphical representation. This provides a unifying framework of building large-scale commonly proposed multivariate statistical models (Kalman filters, hidden Markov models) . Graph Theory has inherent capability to represent dependence and independences of the variables.

Graphs representation is very flexible in a way that it doesn't require to have all the knowledge 0f world for building of a model. One can start his model with his perception of knowledge it has. When one gathers more knowledge it can be incrementally added to the model (Add/Update nodes, edges) the same way and model in turns give improved results based on new information.

As PGMs are standard mathematical structure that not only allows to encode probability distribution and provides a very clear interface for query modelling of prediction queries.

From an abstract point of view, in a graphical model, the joint distribution P_θ is expressed by means of an underlying graph. The nodes and edges in this graph shows random variables and probabilistic relationships between variables. The idea is to present a complex distribution involving a random variables of large number as a product of local functions, where every variable depends only on related variables of a small number, according to the specific independence assumptions that have been done.

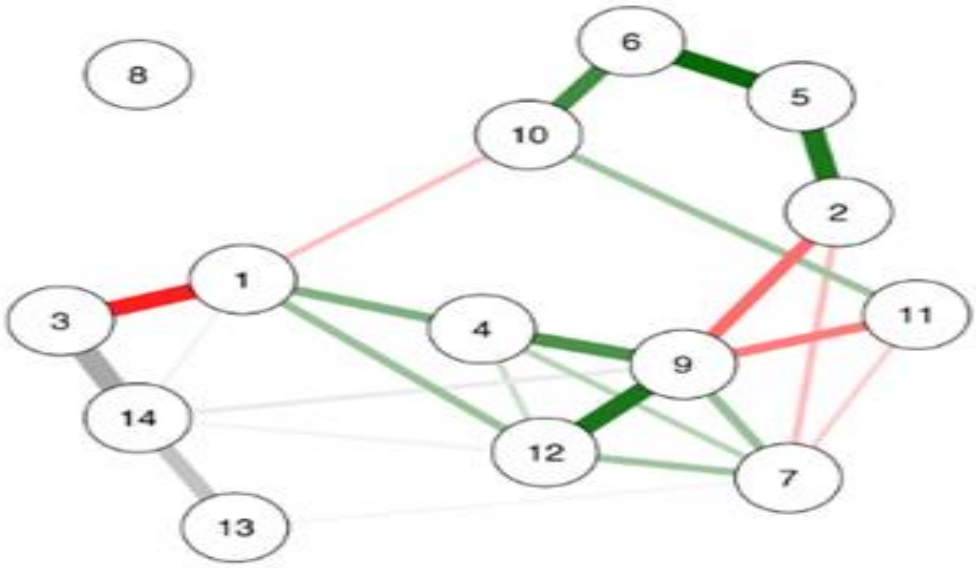


Fig 3.1 14 -States Graphical Model Representation

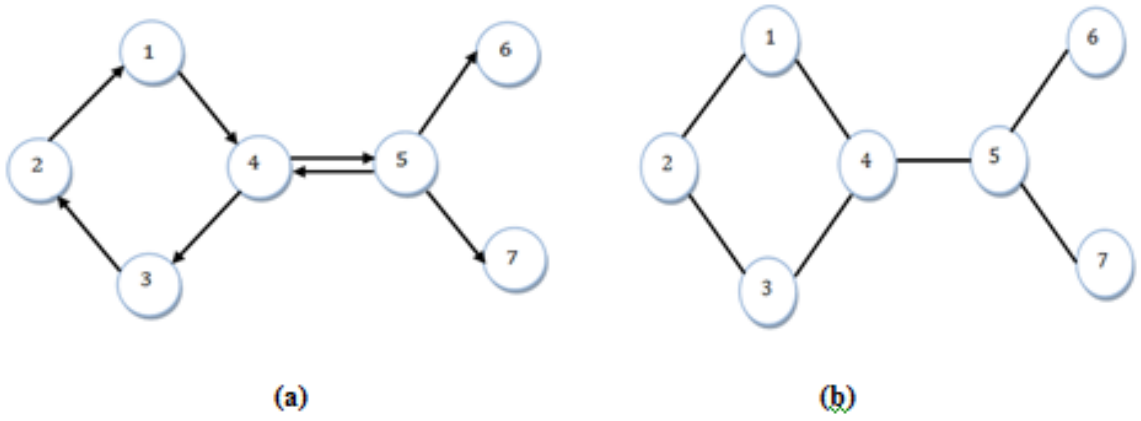
3.1 Types

Probabilistic graphical models uses a graph generally based for encoding complete distribution over a multidimensional arena. Normally known as compact representation of independencies incorporated in specific underlying distribution. Graphical models covers the facts associated with factorization and independences. The known difference lies in way the gets encoded with factorization within distribution.

Whenever we need to perform conditional dependencies modeling Directed graphs are appropriate. On other side , undirected models are suitable for data modeling in which conditional dependencies doesn't exist.

3.1.2 Directed and Undirected Models

Bayesian models come under class of Directed graphs and Markov models are example of undirected graphs. Each of them, models the data set with conditional dependencies between variables. The congruity rest in the element that, how two of them coax between conditional independence between variables. Two examples are given below.



(a) (b)

Fig 3.2 An example of (a) directed graph and (b) undirected graph

3.1.3 Representing Multivariate Distribution

Probabilistic Graphical Models provides more intuitive tools for dealing with multivariate probabilistic models. Such models are defined by variables joint probability P $(W_1, W_2, \dots, W_n)$.

Probabilistic Graphical Model is to represent such joint probabilities in terms of conditional probabilities. It can be rewritten as:

$$P(W_1, W_2, \dots, W_n) = P(W_1)P(W_2|W)P(W_3|W, W_2), \dots \dots, P(W_n|W_1, W_{n-1}) \qquad (3.2)$$

The above equation assumes no preliminary independency information on data. In case of complete independency of models random variables, joint probability is defined as

$$P(W_1, W_2, \dots, W_n) = P(W_1)P(W_2)P(W_3) \dots P(W_n) \qquad (3.3)$$

3.1.4 Markov Networks

Markov Networks comes under the umbrella of Undirected Graphs .The nodes in a Markov network graph corresponds to variables. The edges shows some direct probabilistic interaction between the neighboring variables. A Markov network is thus a graphical way of showing joint probability distribution of random variables.

Lets assume that there are two events that causes the grass to be wet: one can be sprinkler and second is raining. Also, suppose that whenever it is, the sprinkler is usually not turned on.

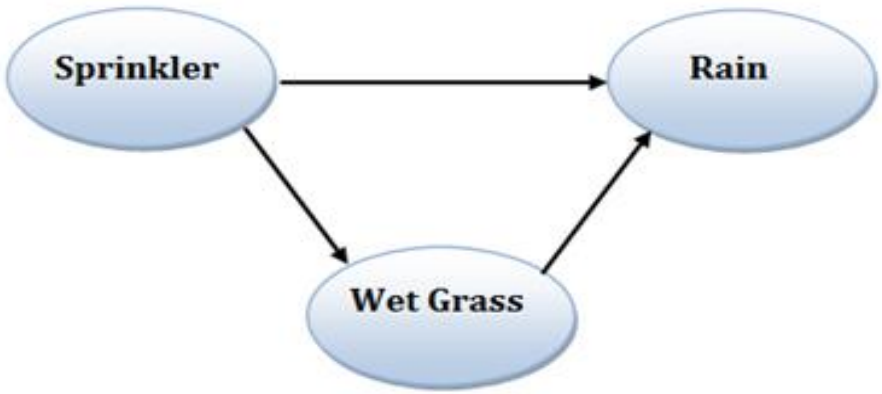


Fig 3.3 Example of Markov process

We can see that this graph is fully connected from one node to another.

3.1.5 Conditional Independence for Markov network

Node y_i is independent (conditionally) of all other nodes present in network given its Markov Blanket that is set of all the neighbors of yi.

3.1.6 Bayesian Networks

Bayes Network represents a probability distribution through DAG (directed acyclic graph). This graph shows conditional dependency. Each of the node shows a unique random variable to which it connects. For example, scenario shown in Fig 3.4WetGrass has an edge coming from rain, which means that it has factor a P(WetGrass|Rain) . For this factor there are specified probability values that leads towards next Wet Grass node in conditional probability table.

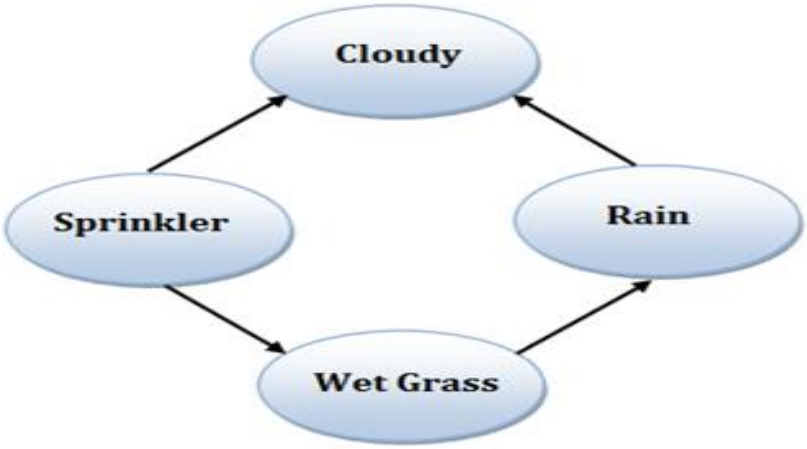


Fig. 3.4 Example of Bayesian Network

CHAPTER 4

Gaussian distribution

Gaussian distribution also known as Normal distribution i.e. bell shaped continuous symmetric curve having unity variance with a center at zero. This distribution finds its usefulness because of central limit theorem.

Central Limit Theorem

During sample distribution, standardized sample mean approaches the standard normal distribution as the sample size keeps on increasing $n \rightarrow \infty$.

$$P(a \leq Z_n \leq b) \approx P(a \leq \phi \leq b) \tag{4.1}$$

Where Z_n denotes standardized sample mean and ϕ is standard normal distribution.

4.1 Univariate case

In case of single variable Y follows normal distribution variance σ^2 and mean μ its pdf will be

$$p(y) = N(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(y-\mu)^2}{2\sigma^2}} (-\infty < y < \infty, y \in R) \tag{4.2}$$

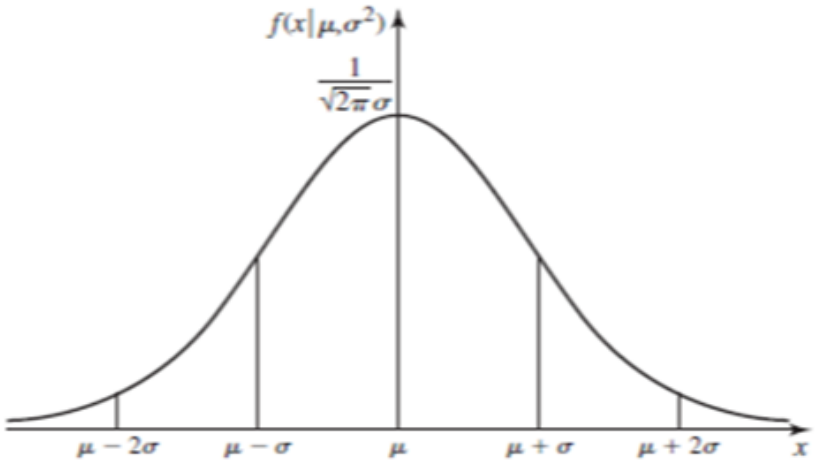


Fig 4.1 Illustration of Univariate Gaussian distribution in one-dimensional space.

Where,

$$\mu = E(y) = \int_{-\infty}^{\infty} y\, p(y)\, dy \tag{4.3}$$

$$\sigma^2 = E(y-\mu)^2 = \int_{-\infty}^{\infty} (y-\mu)^2 p(y)\, d(y) \tag{4.4}$$

4.2.Bivariate case

Extending towards higher K-dimensional variable Y, covariance matrix Σ and mean μ.

The joint p.d.f. of (Y,Z) is

$$f_{Y,Z}(y,z) = \frac{1}{2\pi\sigma_Y\sigma_Z\sqrt{1-\rho}} \exp -\left\{ -\frac{.1}{2(1-\rho^2)} \left[\left(\frac{y-\mu_Y}{\sigma_Y}\right)^2 + \left(\frac{z-\mu_Z}{\sigma_Z}\right)^2 - 2\rho \left(\frac{y-\mu_Y}{\sigma_Y}\right)\left(\frac{z-\mu_Z}{\sigma_Z}\right) \right] \right\} \tag{4.5}$$

where $-\infty < y < \infty$, $-\infty < z < \infty$. Then Y and Z are the bivariate normal distributions and $\rho$is the correlation between them.
The joint generating function for Y and Z is

$$M(t_1 t_2) = \exp\left[t_1\mu_Y + t_2\mu_Z + \frac{1}{2}(t_1^2\sigma_Y^2 + 2\rho t_1 t_2 \sigma_Y\sigma_Z + t_2^2\sigma_z^2) \right] \tag{4.6}$$

4.2.1.Case 1 marginal pdf's of Y and Z;

The moment generating function of X can be given by

$$M_Y(t) = M(t,0) = \exp\left[\mu_Y t + \frac{1}{2}(\sigma_Y^2 t^2) \right] \tag{4.7}$$

Similarly, the moment generating function of Y can be given by

$$M_Z(t) = M(0,t) = \exp\left[\mu_Z t + \frac{1}{2}(\sigma_Z^2 t^2) \right] \tag{4.8}$$

Thus, Y and Z are both marginally normal distributed, i.e.,$Y \sim N(\mu_Y, \sigma_Y^2)$ and $Z \sim N(\mu_Z, \sigma_Z^2)$

Y's pdf is

$$f_Y(y) = \frac{1}{\sqrt{2\pi}\,\sigma_Y} \exp\left[-\frac{(y-\mu_Y)^2}{2\sigma_Y^2}\right] \tag{4.9}$$

Pdf of Z is

$$f_Z(Z) = \frac{1}{\sqrt{2\pi}\,\sigma_Z} \exp\left[-\frac{(z-\mu_Z)^2}{2\sigma_Z^2}\right] \tag{4.10}$$

4.2.2.Case 2 Y and Z are considered independent if ρ =0. (Here ρ is the correlation coefficient)

Ifρ=0 , then

$$M(t_1,t_2) = \exp\left[\mu_Y t_1 + \mu_z t_2 + \frac{1}{2}(\sigma_Y^2 t_1^2 + \sigma_z^2 t_2^2)\right] = M(t_1,0).M(0,t_2) \tag{4.11}$$

Assuming Y and Z to be independent, so

$$M(t_1,t_2) = M(t_1,0).M(0,t_2) = \exp\left[\mu_Y t_1 + \mu_z t_2 + \frac{1}{2}(\sigma_Y^2 t_1^2 + \sigma_z^2 t_2^2)\right]$$
$$= \exp\left[t_1\mu_Y + t_2\mu_Z + \frac{1}{2}(t_1^2\sigma_Y^2 + 2\rho t_1 t_2 \sigma_Y\sigma_Z + t_2^2\sigma_z^2)\right] \tag{4.12}$$

Therefore, ρ = 0

4.2.3.Case 3 Distribution of(Y + Z).

$$M_{Y+Z}(t) = E[e^{t(Y+Z)}] = E[e^{tY+tZ}]$$

Recall that $(t_1, t_2) = E[e^{t_1Y+t_2Z}]$, therefore we can obtain $M_{Y+Z}(t)$by $t_1 = t_2 = t$ in $M(t_1, t_2)$, i.e.

$$M_{Y+Z}(t) = M(t,t) = \exp\left[\mu_Y t + \mu_z t + \frac{1}{2}(\sigma_Y^2 t^2 + 2\rho\sigma_Y\sigma_Z t^2 + \sigma_z^2 t^2)\right]$$
$$= \exp\left[(\mu_Y + \mu_z)t + \frac{1}{2}(\sigma_Y^2 + 2\rho\sigma_Y\sigma_Z + \sigma_z^2)t^2\right] \quad (4.13)$$
$$\therefore Y + Z \sim N(\mu = \mu_Y + \mu_z, \sigma^2 = \sigma_Y^2 + 2\rho\sigma_Y\sigma_Z + \sigma_z^2)$$

4.2.4.Case 4 Conditional pdf of f(y|z), andf(z|y)

Conditional distribution of Y given Z=z is

$$f(y|z) = \frac{f(y,z)}{f(z)} = \frac{1}{\sqrt{2\pi}\,\sigma_Y\sqrt{1-\rho^2}} \exp\left\{-\frac{(y - \mu_Y - \frac{\sigma_Y}{\sigma_Z}(z-\mu_z)^2}{2(1-\rho^2)\sigma_Y^2}\right\} \quad (4.14)$$

Similarly, we have the conditional distribution of Z given Y=y is

$$f(z|y) = \frac{f(y,z)}{f(z)} = \frac{1}{\sqrt{2\pi}\,\sigma_z\sqrt{1-\rho^2}} \exp\left\{-\frac{(z - \mu_z - \frac{\sigma_Z}{\sigma_Y}(y-\mu_Y)^2}{2(1-\rho^2)\sigma_Z^2}\right\} \quad (4.15)$$

Therefore,

$$Y|Z = z \sim N\left(\mu_Y + \rho\frac{\sigma_Y}{\sigma_Z}(z-\mu_z), (1-\rho^2)\sigma_Y^2\right) \quad (4.16)$$

$$Z|Y = y \sim N\left(\mu_z + \rho\frac{\sigma_z}{\sigma_Y}(y-\mu_Y), (1-\rho^2)\sigma_z^2\right) \quad (4.17)$$

4.3 Multivariate case

The multivariate Gaussian distribution for a vector **z** having D-dimensions is defined as:

$$f_Z(z_1, \ldots, z_n) = N(z|\,\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} e^{-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)} \quad (4.18)$$

Where μ is mean vector, Σ is a covariance matrix , defined as

$$\Sigma = E[(z-\mu_i)(z-\mu_i)^T] \quad (4.19)$$

|Σ| denotes the determinant ofΣ and E[.]is mean value of random variable.

For n=2 (recall bivariate normal case) , we have

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \quad , \sigma_{12} = \mathrm{Cov}(Z_1, Z_2)$$

$$\Sigma^{-1} = \frac{1}{\sigma_1^2\sigma_2^2(1-\rho^2)} \begin{bmatrix} \sigma_1^2 & -\rho\sigma_1\sigma_2 \\ -\rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} = \frac{1}{1-\rho^2} \begin{bmatrix} 1/\sigma_1^2 & -\rho/\sigma_1\sigma_2 \\ -\rho/\sigma_1\sigma_2 & 1/\sigma_2^2 \end{bmatrix} \qquad (4.20)$$

Joint density of Z_1 and Z_2, will be

$$f_{Z_1,Z_2}(z_1, z_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho}} \exp\left\{ -\frac{1}{2(1-\rho^2)} \left[\left(\frac{z_1-\mu_1}{\sigma_1^2}\right)^2 + \left(\frac{z_2-\mu_2}{\sigma_2^2}\right)^2 - 2\rho\left(\frac{z_1-\mu_1}{\sigma_1}\right)\left(\frac{z_2-\mu_2}{\sigma_2}\right) \right] \right\} \qquad (4.21)$$

Example: Plotting bivariate pdf of $Z = \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}$, mean $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and covariance $\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

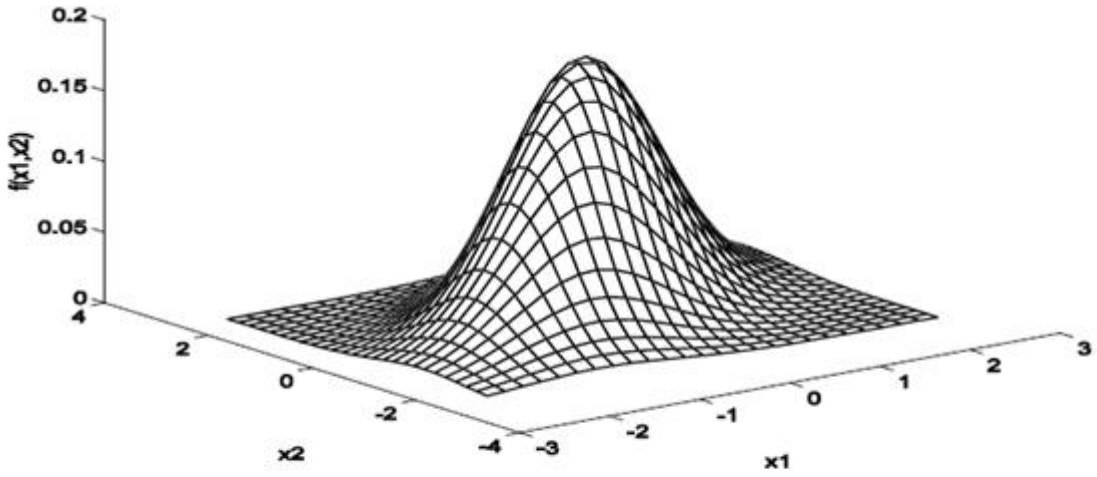


Fig 4.2 Multivariate Normal Distribution

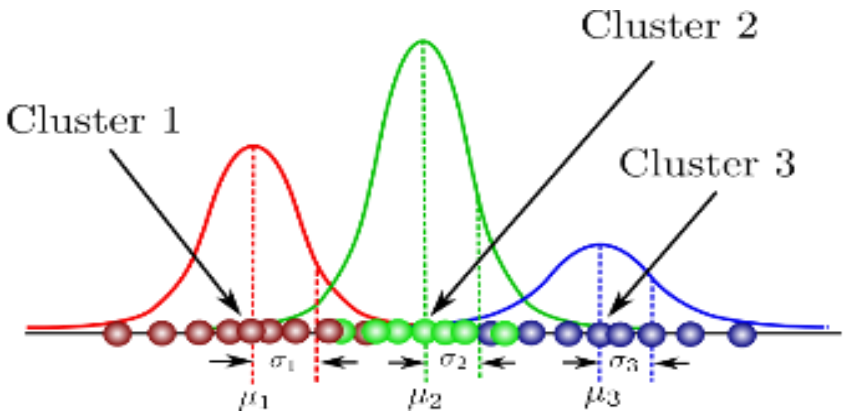
4.4.Properties of Multivariate Normal Distribution

- If $Z_{P\times1} \sim N_P(\mu, \Sigma)$ then Z_j is $N(\mu_j, \sigma^2{}_j)$ for all $Z_j, j = 1,2, \dots, p$.
- If $Z_{p\times1} \sim N_p(\mu, \Sigma)$ then subset of $Z_{p\times1}$ i. e $Z_{q\times1}$ is $N_p(\mu, \Sigma)$.
- If $Z_{p\times1} \sim N_p(\mu, \Sigma)$ then linear combinations of $Z_j, j = 1,2, \dots, p$ is univariate normal.
- If $Z_{p\times1} \sim N_p(\mu, \Sigma)$ then q linear combination of $Z_j, j = 1,2, \dots, p$ is multivariate normal.
- If $X_{p\times1} \sim N_p(\mu, \Sigma)$ then q linear combination of $X_j, j = 1,2, \dots, p$ is multivariate normal.

4.5.Gaussian Mixture Model (GMM)

Gaussian Mixture Model is a type of probabilistic model that under takes the information (number of Gaussians) belonging to a mixture distribution . Each Gaussian 'k' in the mixture has following parameters:

- A center defined by μ (mean).
- Width defined by Σ(a covariance).
- Gaussian function (big or small) defined by π (mixing probability).Illustrating these parameters graphically , with three Gaussian functions (K=3):



The probabilities i.e. mixing coefficients, must meet the following condition:

$$\sum_{k=1}^{K} \pi_k = 1 \qquad (4.22)$$

The Gaussian density function is

$$N(x|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|)^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) \quad (4.23)$$

here **x** shows data points and D represents number of dimensions of every data point. Considering a dataset consisting of N = 1000 with D = 3, and **x is** 1000 × 3 matrix, μ is a 1 × 3 vector, and Σ represents a 3 × 3 matrix. We also have found it beneficial to take the logarithm of equation():

$$\ln\ N(x|\mu,\Sigma) = -\frac{D}{2}\ln 2\pi - \frac{1}{2}\ln\Sigma - \frac{1}{2}\ (x-\mu)^T\Sigma^{-1}(x-\mu) \quad (4.24)$$

Differentiating, equation() w.r.t to the covariance and mean ,equating it to the zero, this will enable us to find these optimal parameter values. The resulting solution, will correlate with Maximum Likelihood Estimates (MLE). As we are dealing with many Gaussians instead of one, things gets a little complicated, then there comes a time to find whole mixture parameters.

4.5.1.Initial derivations

First, from k Gaussian, the probability of x_n(data point), is shown below

$$p(z_{nk} = 1|x_n) \quad (4.25)$$

Here, z is a latent variable and it takes two possible values only i.e. first is **x** that comes from Gaussian k, and zero otherwise[22]. Knowing the probability of occurrence of variable z will be helpful in determining the Gaussian mixture parameters. However, when we state the following:

$$\pi_k = p(z_k = 1) \quad (4.26)$$

So the overall probability of observation of a point coming out from Gaussian k, is basically equal to the mixing coefficients for that Gaussian, i.e. the bigger the Gaussian, then higher probability expectation. Now, suppose **z** contains all possible latent variables $z_{1\ldots\ldots}z_k$.

$$\mathbf{z} = z_{1\ldots\ldots}z_k \quad (4.27)$$

Here each z occurs independently of others. Therefore:

$$p(\mathbf{z}) = p(z_1 = 1)^{z_1}\, p(z_2 = 1)^{z_2} \ldots . p(z_K = 1)^{z_K} = \prod_{k=1}^{K} \pi_k^{z_k} \qquad (4.28)$$

On seeing that the probability of observing our data given that it came from Gaussian k, turns out to be that it is actually the Gaussian function. We can state:

$$p(x_n|\mathbf{z}) = \prod_{k=1}^{K} N(x_n|\mu_k, \Sigma_k)^{z_k} \qquad (4.29)$$

Our initial aim was to determine the probability of z given our observation **x**, Bayes rule, will help us to determine this probability. From the product rule of probabilities, we have;

$$p(x_n, \mathbf{z}) = p(x_n|\mathbf{z})p(\mathbf{z}) \qquad (4.30)$$

So we will firstly need **p(**x_n**)**, not **p(**x_n, **z).**We will be using Marginalization property to get rid of **z**. Hence by summing up the terms on **z**, we get

$$p(x_n) = \sum_{k=1}^{K} p(x_n|\mathbf{z})p(\mathbf{z}) = \sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k) \qquad (4.31)$$

This equation 4.31 defines a Gaussian Mixture Model , and it mainly depends on all parameters that were previously mentioned. The maximum likelihood for the model needs to be determined to determine these optimal values. The likelihood can be found as joint

probability of all observationsx_n

$$p(\mathbf{X}) = \prod_{n=1}^{N} p(x_n) = \prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k) \qquad (4.32)$$

Again taking log on each side of the equation (4.32)

$$\ln p(\mathbf{X}) = \prod_{n=1}^{N} p(x_n) = \sum_{n=1}^{N} \ln \sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k) \qquad (4.33)$$

Maximizing this function turns out to be more complex problem because of summation over k inside logarithm [23].

By setting the log likelihood derivatives to zero, we will not get a closed form solution. Hence we will use another technique which is known as Expectation Maximization EM.

4.5.2.Expectation Maximization Algorithm

Expectation maximization method is used for maximum likelihood solutions for latent variables model.

From 3.13, we have

$$\ln p(X|\mu, \pi, \Sigma) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k) \right\} \qquad (4.34)$$

By taking derivative of above Equation (4.34) w.r.t mean μ_k and equate it to zero. We have,

$$\frac{\partial}{\partial \mu_k} \left(\sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k) \right\} \right) = 0 \qquad (4.35)$$

$$\sum_{n=1}^{N} \frac{\pi_k \frac{\partial}{\partial \mu_k} N(x_n|\mu_k, \Sigma_k)}{\sum_{l=1}^{K} \pi_l N(x_n|\mu_l, \Sigma_l)} = 0 \qquad (4.36)$$

By simplifying we get,

$$\sum_{n=1}^{N} \frac{\pi_k}{\sum_{l=1}^{K} \pi_l N(x_n|\mu_l, \Sigma_l)} \times 1/2\pi^{\frac{2}{2}}\Sigma^{\frac{1}{2}} \times e^{-\frac{1}{2(x-\mu)\Sigma^{-1}(x-\mu)^T}} \left(-\frac{2}{2}\right)(x_n - \mu_k)(-1) = 0 \qquad (4.37)$$

and

$$\sum_{n=1}^{N} \gamma(z_{nk}) \times \frac{1}{\Sigma_k} \times (x_n - \mu_k) = 0 \qquad (4.38)$$

$$\sum_{n=1}^{N} \gamma(z_{nk}) x_n = \sum_{n=1}^{N} (\mu_k) \qquad (4.39)$$

This gives the value of mean

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n \qquad (4.40)$$

By fixing the derivative $\ln p(X|\mu, \Sigma, \pi)$ w.r.tΣ_k=0.By following same line of reasoning, we obtain the result,

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) (x_n - \mu_k)(x_n - \mu_k)^T \qquad (4.41)$$

At the end, maximizing$\ln p(X|\mu, \Sigma, \pi)$ with respect to $\pi_k$that is a mixing coefficient and taking into account constraint 3.12, which requires mixing co-efficient to some to 1.It can be accomplished by a Lagrange multiplier and maximizing the following quantity and equate it to zero

$$\ln p(X|\mu, \pi, \Sigma) + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right) \qquad (4.42)$$

By simplifying,

$$0 = \sum_{n=1}^{N} \frac{N(x_n|\mu_k,\Sigma_k)}{\sum_{l=1}^{K} \pi_l N(x_n|\mu_l,\Sigma_l)} + \lambda \qquad (4.43)$$

Multiplying both sides by $\pi_k$we obtain:

$$0 = \sum_{n=1}^{N} \gamma(z_{nk}) + \lambda \pi_k \qquad (4.44)$$

Which gives

$$0 = N_k - N\pi_k \qquad (4.45)$$

Hence,

$$\pi_k = \frac{N_k}{N} \qquad (4.46)$$

4.5.2.1.EM algorithm for GMM

Given a GMM, our goal is to maximize the likelihood function with respect to the parameters comprising mixing coefficients, means and covariances of components .

Step-1

Initializing the mean μ_j, covariance Σ_j and mixing coefficient π_j. Then evaluating the initial values of the log likelihood where $j = 1,2,3, \dots, k$.

Step-2

Evaluating responsibilities using current parameter values for E-step.

$$Y_k(x) = \frac{\pi_k N(x|\mu_k \Sigma_k)}{\sum_{j=1}^{K} \pi_j N(x|\mu_j, \Sigma_j)} \qquad (4.47)$$

where $Y_k(x)$ is latent variable for k^{th} Gaussian.

Step-3

Re-calculating the parameters by using current obtained values for M-step

$$\mu_j = \frac{\sum_{n=1}^{N} Y_{j\,(x_n)} x_n}{\sum_{n=1}^{N} Y_j\,(x_n)} \tag{4.48}$$

$$\Sigma_j = \frac{\sum_{n=1}^{N} Y_j(x_n)(x_n - \mu_j)(x_n - \mu_j)'}{\sum_{n=1}^{N} Y_j(x_n)} \tag{4.49}$$

$$\pi_j = \frac{1}{N}\sum_{n=1}^{N} Y_j(x_n) \tag{4.50}$$

Step-4

Evaluating the Log-likelihood,

$$\ln p(X|\mu, \Sigma, \pi) = \sum_{n=1}^{N} \ln\left\{\sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k)\right\} \tag{4.51}$$

We have to find out that if these parameters truly represent data points. If this does not happen, go back to step-2, which will help to converge to better solution.

CHAPTER 5

Hidden Markov Model

Baum and Petrie introduced HMMs in 1960[25].HMMs are the statistical stochastic model sequence. They can be used in the analysis of speech synthesis ,cryptoanalysis,reinforcement learning, temporal pattern recognition, gesture recognition, and part-of-speech tagging.

In HMM, states cannot be observed directly but they can get identified through vector series observation. Since 1980s, HMM has been successfully incorporated in speech recognition, mobile communication and image processing techniques. The basic principle followed in HMM is that the events observed have no direct affinity with states. Thus, links find their connection to the states by the probability distribution.

Markov Chain classifies stochastic processes and state transitions, narrating the statistical congruity between observed values and the states. From the stance of observers, only the observed point can be perceived but the states may not.

Therefore, a Hidden Markov Model is a stochastic procedure that identifies the presence of states with their characteristics. HMM is divided into two sections . One , has found application in Markov Chain.Secondly, runs the necessary algorithms to describe HMM for problem solving in computer sciences.

5.1. Markov Chain

A Markov Chain is described by a set of state space variables $S = \{s_1, s_2, \dots\dots, s_N\}$ with probability $P_r(S_i(t))$ at any time t, with state i i.e.$s_{i1}, s_{i2}, \dots, s_{ik}$ .... Dynamics of Markov Model are specified by transition probabilitya_{ij} and initial probability $\pi_i = P(S_i)$.This means that probability of present state at time t is S_j given that it is currently present in state S_i at state (t-1) for $(S_i, S_j) \in S$. This leads to Markovanian assumption a_{ij} (transition probability) is applied whenever state S_i is being visited by previous independent states to reach S_i.This is a Markov Chain property statement

$$a_{ij}(t-1,t) = P_r\left(S_j(t)\middle|S_i(t-1), \dots\dots\dots S_1(0)\right) = P_r\left(S_j(t)\middle|S_i(t-1)\right)$$
$$\forall\, t \text{ and } (S_i, S_j) \in S \qquad (5.1)$$

In general $a_{ij}(t-1,t)$ is dependent on time difference i.e. if the chain is time homogeneous , resulting in stationary transition probability , such that $a_{ij}(t-1,t) \rightarrow a_{ij}$ with $a_{ij} \geq 0$ and $\sum_{n=1}^{N} a_{ij} = 1$.

In$N_s \times N_s$, where homogeneous chain elements are embedded , state transition probability matrix $A = [a_{ij}]; i,j = 1 \ldots . . N_s$. Fig 4.1.

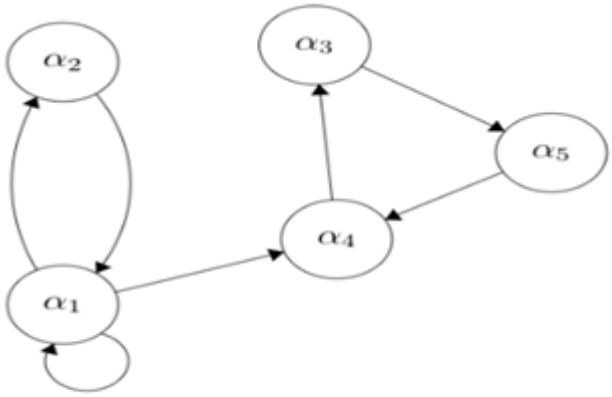


Fig 5.1 Directed graph representation of five-state Markov chain

5.2 Hidden Markov Model

In Markov Chains, transition states are directly noticeable to observer. However, in HMM, desired state is invisible only output dependent state is available for stochastic modelling. Hidden Markov Model is described as a stochastic measurement of Markov Chain or as GMM variable changing with time.

5.2.1. From observable to hidden state

HMMs provide great help when there is need of modeling a process in which we do not have direct knowledge about the present state of system. The only direct knowledge that we have about the process to model is the set of observations it generates while we don't have direct access to the internal structure of process. It is easier to build a model that gives good approximation of process, when we have specific knowledge of the domain, but, in many cases this problem doesn't have easy solution and it can be task-dependent.

5.2.2 HMM Parameters

In Markov Model, output symbols and states are equal. Hence, the observation is comparable with the state. In Hidden Markov Model we have no information of current state . Following equation gives us possible tag sequence.

$$\hat{t}_n^1 = \text{argmax}_{t_n^1} P(t_n^1 | w_n^1) \quad (5.2)$$

Where t_n^1 and w_n^1denotes speech tag sequence with sequence of words from $1,2,\dots,n$ respectively. Bayes rule is usedfor transforming the equation (5.2) into other probabilities set .

$$\hat{t}_n^1 = \text{argmax}_{t_n^1} \frac{P(w_n^1 | t_n^1)\, P(t_n^1)}{P(w_n^1)} \quad (5.3)$$

Observation sequence length	L
Number of states	D
Observation numbers	H
States	$Q = (q_1, q_2, \dots, q_n)$
Possible symbols	$V = (v_1, v_2, \dots, v_m)$
Transition Probability Matrix	$A = \{a_{ij}\}$
Probability Matrix Output	$B = \{b_j(H_k)\}$
Initial State Vector	π

Table 5-1 Hidden Markov Model Parameters

5.3 Example

For any non-trivial task whenever we are asked to do a task, we find that information that we have to work with is very much partial. In such cases we have to deal with uncertain information. Let's suppose that we have three urns and each of them contain Red, Green, Yellow and Blue balls as shown in fig.

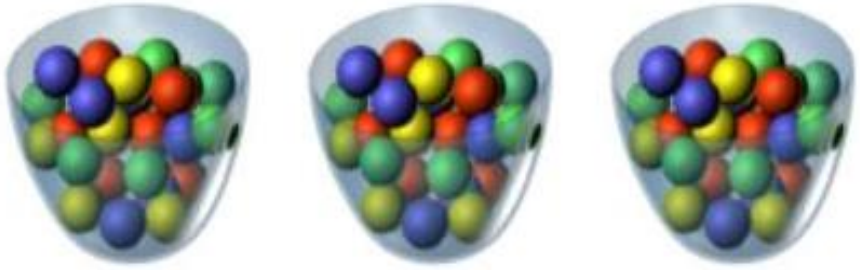
A person is picking ball from these urns and it gives us a pattern of drawing a ball from these urns as YRGBYYBB.

We need to find out the sequence of urns from which he had drawn the balls. Hence, for this sequence of colors of balls, we have to produce urns sequence or state sequence that is given as,

$$\{U_1, U_2, U_3, \ldots, U_i\} \tag{5.4}$$

We have information of probability of transition from one urn to any other urn. Suppose we drew a ball out of urn 1 and again he drew a ball out from urn 1 is 0.1. Table (5.2)consists of probabilities of specific color balls in urn. As we got the information about the number of balls with given color probabilities . These two things are known and by using these, we can compute the probable state sequence which is hiddens. We are given

Observation Sequence= YRGBYYBB



# of Red = 30	# of Red = 10	# of Red = 60
# of Green = 50	# of Green = 40	# of Green = 10
# of Blue = 20	# of Blue = 50	# of Blue = 30
# of Yellow = 10	# of Yellow = 30	# of Yellow = 20

Fig.5.2 Three Urns Containing Red, Green, Blue and Yellow balls

	U1	U2	U3
U1	0.1	0.4	0.5
U2	0.6	0.2	0.2
U3	0.3	0.4	0.3

Table 5-2 (a) Probability of transition

	R	G	B	Y
U1	0.3	0.5	0.2	0.6
U2	0.1	0.4	0.5	0.3
U3	0.6	0.1	0.3	0.5

Table 5-2 (b) Probability of drawing a ball

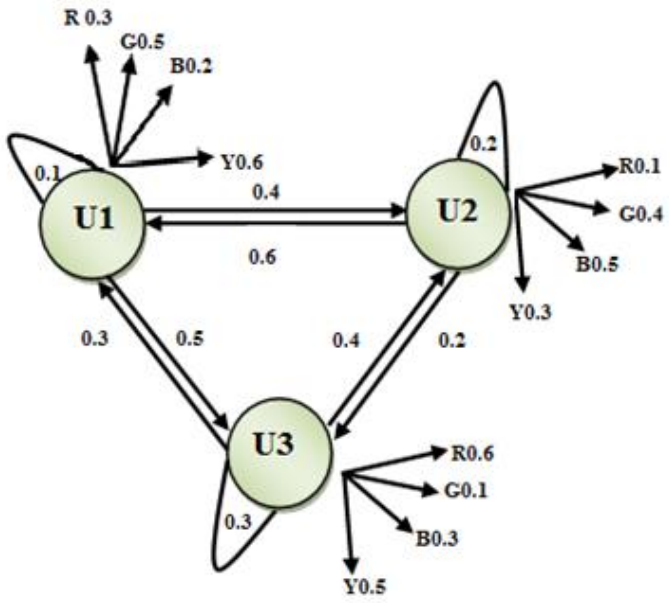


Figure 5.3 Diagrammatic Representation

The data is aggregated as

$$D = \{U1, U2, U3\}$$

$$V = \{R, G, B, Y\}$$

$$H = \{h_1, h_2, \dots, h_n\}$$

$$Q = \{q_1, \dots, q_n\}$$

$$\pi_i = P(q_1 = U_1)$$

	H_1	H_2	H_3	H_4	H_5	H_6	H_7	H_8
Observations	R	R	G	G	B	R	G	R
States	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8

Table 5-3 Observations and States

The above table shows the pattern of drawing balls as RRGGBRGR with observations $H_1, H_2, \dots, H_8$ and set of states $D_1, D_2, \dots, D_8$. From the given table we can see that we need to find out most probable state sequence that is hidden to us. Hence we need to maximize P(D | H)

$$D^* = \text{argmax}_s\big(P(D|H)\big) \qquad (5.5)$$

So this paves way to the efficient calculation because Markov presumption takes highest probability tree leaves. Applying Markov Chain on P(D | H) gives

$$P(D|H) = P(D_1|H)P(D_2|D_1, H)P(D_3|H_2, H), \dots, P(D_8|D_7, H) \qquad (5.6)$$

These terms of probability are difficult to solve because of other clumsy items needs to be fixed. Bayes' theorem plus Markov assumption serves this purpose.

5.4 Hidden Markov Model Essential Features

- **Naïve Bayes + Markov Assumption**

  Bayes theorem plus Markov assumption is a powerful kit for solutions of problem in machine learning. Incorporating the theorem, the observation sequence with state transition probabilities will be discussed below.

- **State Transitions Probability**

  Preceding P(S) is used in the way

$$P(D) = P\ (D_{1-8}) \tag{5.7}$$

$$P(D) = P(D_1)\ P(D_2|D_1)\ P(D_3|D_{1-2}) \dots \dots\ P(D_8|D_{1-7}) \tag{5.8}$$

According to Markov assumption, (k=1)

$$P(D) = P(D_1)P(D_2|D_1)P(D_3|D_2) \dots \dots P(D_8|D_7) \tag{5.9}$$

- **Observation Sequence Probability**

  Next probability item P (H | D) turns to be:

$$P(H|D) = P(H_1|D_{1-8})\ P(H_2|H_1, D_{1-8})\ P(H_3|H_{1-2}, D_{1-8}) \dots\ P(H_8|H_{1-7}, D_{1-8}) \tag{5.10}$$

Assuming the drawn ball is based on urn chosen

$$P(H|D) = P(H_1\ |D_1)\ P(H_2|\ D_2)P(H|\ D_3) \dots \dots P(H_8|\ D_8) \tag{5.11}$$

By applying Bayes' theorem we have,

$$\text{argmax}_s P(D|H) = \text{argmax}_s P(D)\ P(H|D) \tag{5.12}$$

Here the denominator p(H) is ignored because it is independent of S so it can be eliminated from consideration. Hence by putting values in 4.6 we have

$$P(D|H) = P(D_1)\ P(D_2|D)\ P(D_3|D_2) \ldots P(D_8|D_7)\ P(H_1|D_1)\ P(H_2|D_2)$$
$$P(H_3|D_3) \ldots \ldots P(H_8|O_8) \qquad (5.13)$$

All these terms are then grouped together according to the equation (5.14)

$$P(H_k|D_k).\,P(D_{k+1}|D_k) = P\left(D_k \overset{H_k}{\rightarrow} D\right) \qquad (5.14)$$

The diagram shown below shows set of observations with corresponding states.

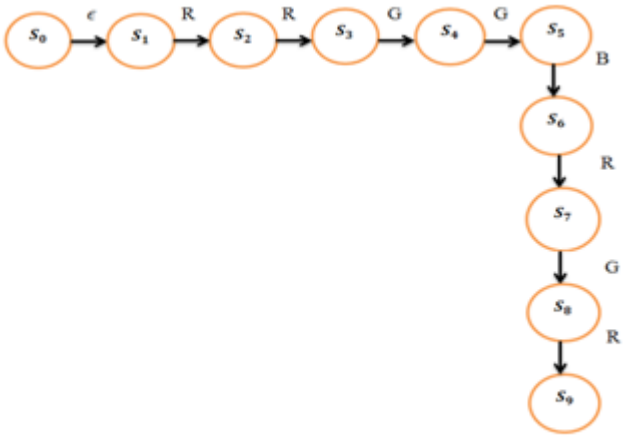


Fig 5.4 Diagrammatic Representation of Observations and State

5.5 Hidden Markov Model Properties

These Markov process properties provides base for HMM.

- **Limited Horizon**

It says preceding **t** states, and a state **i** is independent of prior 0 to $t - k + 1$ states. After k states comes before t_{th} state, everything can be neglected. Hence, this is window property . Procedure is acknowledged to be k Markov process.

$$P(W_t = i\ |W_{t-1}, W_{t-2}, \ldots, W) = P(W_t = i\ |W_{t-1}, W_{t-2}, \ldots, W_{t-k}) \qquad (5.15)$$

- **Time Invariance**

  Dependency of one state on prior state is done by entire sequence of Marko process. Referring to conditional probability below is position invariant i.e. not transitioned from point to point in a sequence.

$$P(W_t = i \mid W_{t-1} = j) = P(W_1 = i \mid W_0 = j) = P(W_n = i \mid W_{n-1} = j) \qquad (5.16)$$

5.6. Probability Laws

There are two essential probability laws.

- **Chain Rule**

  Chain rule breaks complete sequence into set with terms

$$P(W_1, W_2, .., W_k) = P(W_1)\, P(W_2|W_1)\, P(W_3|W_2W_1)\, P(W_k|W_{k-1}W_{k-2} \dots W_1) \qquad (5.17)$$

- **Marginalization**

$$P(A) = \Sigma\, P(A, B_1, B_2, B_3 \dots B_n) \qquad (5.18)$$

Where $(B_1, B_2, B_3 \dots B_n)$has all possible values

5.7. Problems in HMM

In this section we deal with basic problems related to HMMs with theirsolutions through efficient algorithms.

5.7.1. Evaluation Problem

Assuming the model given is represented by$\theta = (A, B, \pi)$ and observation sequences are represented by O, we need to calculate probability for a output sequence produced by model θ.

5.7.2. Decoding Problem

For a given $\theta = (A, B, \pi)$ with observation sequence H, we would find most probable sequence of Hidden states that led to generation of the given set of observations. In other words, we need to stimulate the hidden parts contained in Hidden Markov Model.

5.7.3. Learning Problem

Given a set of output sequences that is set of visible states H and set of hidden states D, we have to find out the set of transition probabilities $a_{ij}$and $b_j(H_k)$.

5.8. Problem 1: Forward & Backward Probability Algorithm

5.8.1. Forward Probability Algorithm

$F(k,i)$ forward probability is the probability of being in state D_i with observations $D_0, D_1, D, \dots, D_k$. M being sequence length

$$F(k,i) = P(H_0, H_1, H_2, \dots, H_k, D_i) \tag{5.19}$$

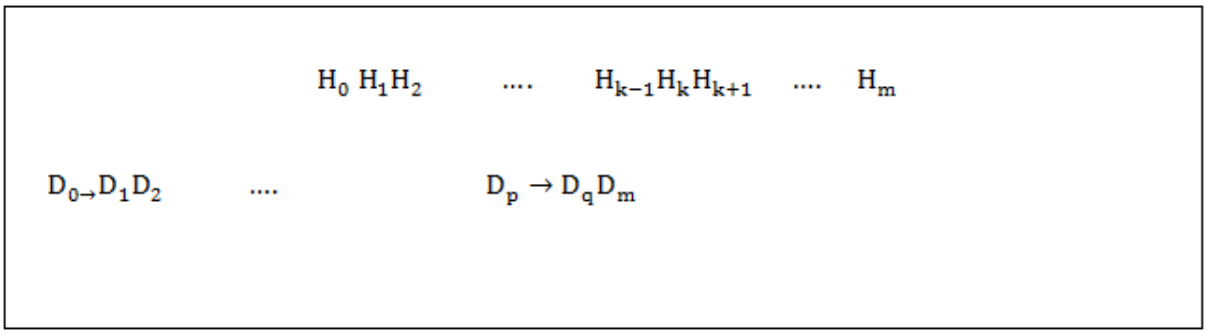
The observed sequence probability is $P(H_0, H_1, H_2, \dots, H_m)$ is sidelined to obtain below equation

$$P(H_0, H_1, H_2, \dots, H_m) = \sum_{i=0}^{N} P(H_0, H_1, H_2, \dots, H_{m_0}, D_p) \tag{5.20}$$

Hence, forward probability is;

$$P(H_0, H_1, H_2, \dots, H_m) = \sum_{i=0}^{N} F(m,p) \tag{5.21}$$

To get $F(m, p)$, we see to sequence $(H_0, H_1, H_2, \dots, H_k)$. We have the observations and states as seen in below figure. Starting with D_0 proceeding to any state with transition state$D_p \rightarrow D_q$ on $H_k$symbol.



So, the forward probability

$$F(k,q) = \left(P(H_0, H_1, H_2, \dots \dots, H_k, D_q)\right) \tag{5.22}$$

$$F(k,q) = \left(P(H_0, H_1, H_2, \dots \dots, H_{k-1}, H_k, D_q)\right) \tag{5.23}$$

Marginalizing,

$$F(k,q) = \sum_{p=0}^{N} \left(P(H_0, H_1, H_2, \dots\dots, H_{k-1}, H_k, D_q)\right) \quad (5.24)$$

By chain rule,

$$F(k,q) = \sum_{p=0}^{N} P(H_0, H_1, H_2, \dots, H_{k-1}, D_p) P(H_k, D_q | (H_0, H_1, H_2, \dots, H_{k-1}, D_p) \quad (5.25)$$

$$F(k.q) = \sum_{p=0}^{N} F(k-1,p) P(H_k, D_q | D_p) \quad (5.26)$$

$$F(k.q) = \sum_{p=0}^{N} F(k-1,p)\ P(D_p \overset{O_k}{\rightarrow} D_q) \quad (5.27)$$

$$F(k,q) = \sum_{p=0}^{N} F(k-1,p) \quad (5.28)$$

Complexity in forward probability lies basically in length of observed sequence is |D| multiplied by length of states |H|. The expression for calculating(k, q)is;

$$T_k = \sum_{i=0}^{N} T_{k-1} \quad (5.29)$$

5.8.1.1. Forward Algorithm Boundary Conditions

Boundary condition is

$$F(0,q) = P_q \qquad (5.30)$$

where P_q is the initial probability present in state D_q that is $D_p \rightarrow D_q$. It is not easy to calculate it in time proportionality w.r.t observation sequence length. Therefore, it is considered as time linear computation.

5.8.2. Backward Probability Algorithm

B (k, i) backward probability is described for seeking symbols $H_k, H_{k+1}, H_{k+2}, \dots, H_m$ with given state D_i.M is the sequence length

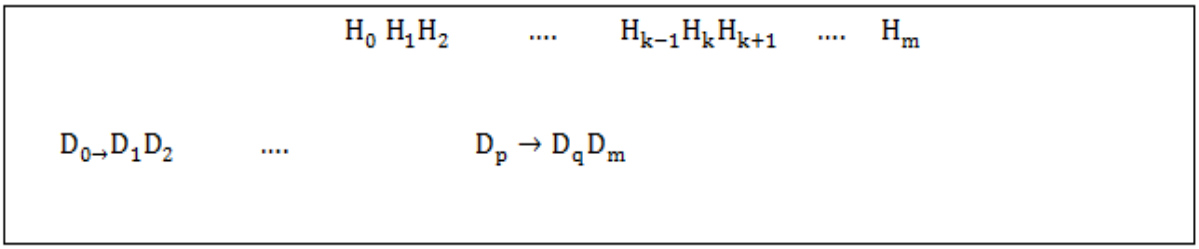
$$B(k,i) = P(H_k, H_{k+1}, H_{k+2}, \dots, H_m | D_i) \qquad (5.31)$$

Observed sequence probability is $P(H_0, H_1, \dots, H_m)$ is marginalized to get following N states. This is single backward probability with 0 as argument

$$P(H_0, H_1, \dots, H_m) = \sum_{i=0}^{N} P(H_0, H_1, \dots, H_m | D_0) \qquad (5.32)$$

$$P(H_0, H_1, \dots, H_m) = B(0,0) \qquad (5.33)$$

Backward probability is computed similarly as forward probability. Again referring back to diagram; the state transition is $D_p \rightarrow D_q$ over symbol H_k.Backward probability is expressed as



$$B(k,p) = P(H_k, H_{k+1}, H_{k+2} \ \ldots, H_m | O_p) \quad (5.33)$$

$$B(k,p) = P(H_k, H_{k+1}, H_{k+2} \ \ldots, H_m, H_k | D_p) \quad (5.34)$$

$$B(k,p) = \sum_{q=0}^{N} P(H_k, H_{k+1}, H_{k+2} \ \ldots, H_m, H_k, D_q | D_p) \quad (5.35)$$

$$B(k,p) = \sum_{q=0}^{N} P(H_k, D_q | D_p) \, P(H_k, H_{k+1}, H_{k+2}, \ldots, H_m | H_k, D_q, D_p) \quad (5.36)$$

$$B(k,p) = \sum_{q=0}^{N} P(H_k, H_{k+1}, H_{k+2} \ \ldots, H_m | D_q) \, . P(H_k, D_q | D_p) \quad (5.37)$$

$$B(k,p) = \sum_{q=0}^{N} B(k+1,q) \, . P\left(D_p \overset{O_k}{\rightarrow} D_q\right) \quad (5.38)$$

For any of the observed sequence with corresponding state sequence, notion k_{th} placed at any point in the stream, enables us to calculate forward probability of a point with backward probability from that point towards observation sequence ending point.

5.8.2.1. Backward Algorithm Boundary Conditions

The (k+1)term keeps increasing until observation sequence ends. So we designate boundary condition for the algorithm.

Mentioning last symbol in entire observation sequence, system is going to be in final state. Thus ,transitioning D_m to $D_{final}$with output asH_m ,it will set boundary condition for the backward algorithm ($D_m \overset{H_m}{\longrightarrow} D_{final}$) . So from last symbol we obtained$B(k,p).D_{final}$is a Hidden Markov Model states.

5.8.3 Problem 2: Viterbi Algorithm

Decoding problem in HMM, seeks to search the optimal sequence state related to given observation sequence H with a given parameter λ , it can be solved using Viterbi algorithm.

The Viterbi algorithm finds the appropriate order of hidden states i.e. Viterbi path. This will give us observed events sequence, especially in structure of HMM and Markov data

source. It is used in keyword spotting ,speech recognition, computational linguistics, bioinformatics and speech synthesis,.

Considering the above HMM example, there are three urns with a need to find state order sequence with specific observation sequence. State sequence is found by Viterbi algorithm, which gets explained with another example mentioned below.

Considering finite state mechanism with S_1 and S_2 (states) and a_1 and a_2 (symbols) with transition states shown in figure below.

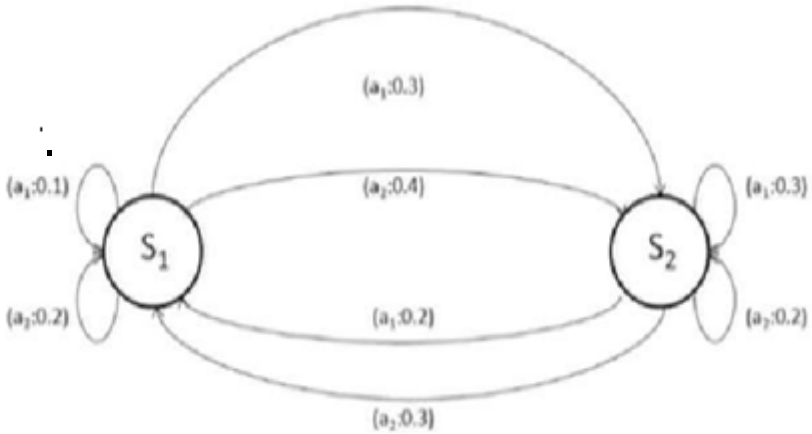


Fig 5.6: Probabilistic Finite State Machine Example

The transition probability from S_1 to S_2 with a_1 at output is 0.4. Probability of being in state S_1 giving a_1 at output is 0.1 and so on. Our goal is to find best possible state sequence path for given observation sequence.

Observation Sequence = $a_1 a_2 a_1 a_2 a_1 a_2$

Problem = $S^* = \text{argmax}_S P(a_1 - a_2 - a_1 - a_2)$

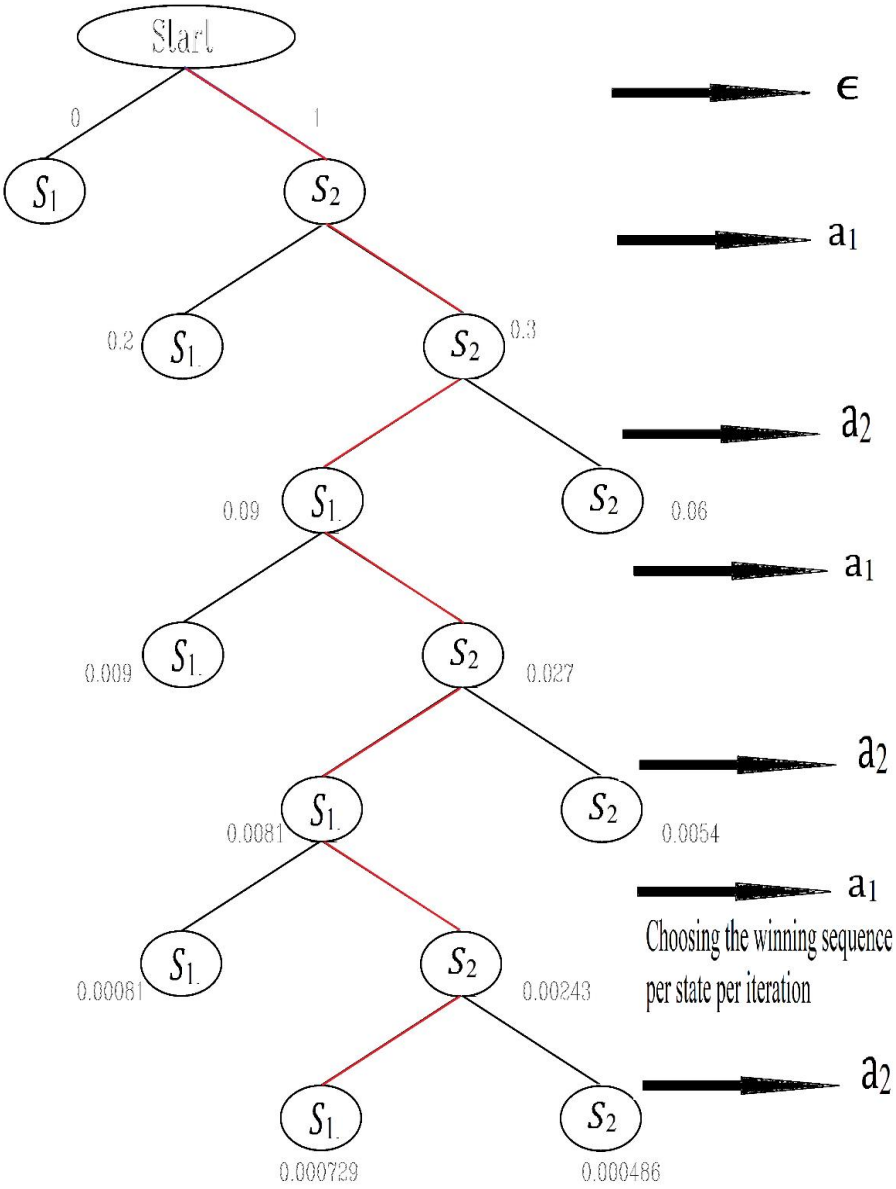


Fig 5.7: Tree Diagram for Viterbi algorithm

Consider two possible states S_1, S_2 in star. Each state has further two states. Transition probability of every state is multiplied with preceding state, to get current state probability and so on. Computed probabilities gets entered in Viterbi tree. This will continue in the direction of highest probability. Remaining states gets cancelled. After completion, highest probability nodes gets tracked backwardly from the end towards the top. The path found is $S_2S_2S_1S_2S_1S_2S_1$.

5.8.3.1. Viterbi Algorithm Steps:

Given:

.

1. HMM
   - Initial state: D_1
   - Alphabet: $A=\{a_1, a_2, \ldots\ldots\ldots\ldots, a_p\}$
   - States: $D = \{D_1, D_2, \ldots\ldots\ldots\ldots, D_n\}$
   - Transition Probability $P(D_i \xrightarrow{a} D_j)$
2. Output String $\{a_1, a_2, \ldots\ldots\ldots\ldots, a_T\}$

To Find:

The most likely state sequence$E_1, E_2, \ldots, E_T$is producedwith given output sequence.

The Data Structure for this algorithm is interpreted as;

Data Structure

- An N*T array called SEQSCORE for maintaining winner sequence

(N= number of states, T= Output sequence)

- Another N*T array called BACKPTR for recovering path.

Step of Viterbi algorithm :

- Initialization
- Iteration
- Sequence Identification

I. Initialization:

SEQSCORE (1, 1) = 1.0

BACKPTR (1, 1) = 0

For (i=2 to N) do

SEQSCORE (i, 1) = 0.0

[Demonstrating that D1 first state]

Step 1 shows that D1 is the starting state and in step 2 we have that there is no state before this. In step 3 we defined that we make probability value 0 in all other states except D1.

II. Iteration

For (t=2 to T) do

For (i=1 to N) do

$$\text{SEQSCORE}(i,t) = \text{Max}_{j=1,N}$$

$$\left[\text{SEQSCORE}\left(j,(t-1)\right).\text{P}\left(D_i \xrightarrow{a_k} D_j\right)\right]$$

$$\text{BACKPTR}\,(I,t) = \text{index j gives maximum of above}$$

In first step we go of our observation sequence symbol by symbol. T is the observation sequence length for every symbol on the observation sequence. In second step we do iteration over the number of states so this is to record the state in which sequence is ending. In third step we have to make sure that we only advance k states at particular level, we do not advance any state whose probability value is less than the winner sequence probability value ending in particular state.

Fourth step shows the multiplication of accumulated sequence probability by the transition probability. Last step is a way of keeping the pointer to be able to recover the state sequence.

III. Sequence Identification

E(T)= i maximizes SEQSCORE (i, T)

For i from (T-1) to 1

$$E(i) = \text{BACKPTR}\,[E(i+1),(i+1)]$$

It shows the state sequence which has found to be the highest probability state sequence.

5.8.4 Problem 3: Baum- Welch (Forward-Backward Algorithm)

This algorithm is applied on training Hidden Markov Model. Thus making an inference that k_{th} hidden variable with $(k-1)_{th}$ given is independent of preceding hidden variables. This shows present observed variable is conditioned upon on present hidden state and is independent from other. Baum–Welch algorithm assimilates EM-algorithm for HMMs maximum likelihood estimates . Considering the following example Baum Welch algorithm having (i) two states, o and q(ii)c and d are symbols.

String = cddcccdddccc

Output Sequence $= o \xrightarrow{c} q \xrightarrow{d} o \xrightarrow{d} o \xrightarrow{c} q \xrightarrow{c} o \xrightarrow{c} q \xrightarrow{d} o \xrightarrow{d} o \xrightarrow{d} o \xrightarrow{c} q \xrightarrow{c} o \xrightarrow{c} q$

Source	Destination	Output	Counts
O	Q	C	6
O	O	D	4
Q	O	C	4
Q	O	D	3

Fig 5-4 Table of counts

Through table (5.4) , we are able to estimate transition probabilities. Assuming $o \xrightarrow{c} q$ transition arises 6 times in output sequence. Total transition numbers from q being initiating state $10(o \to q = 6 + o \to o = 4)$.

Hence,

$$P\left(o \xrightarrow{c} q\right) = \frac{6}{10} \tag{5.39}$$

Likewise,

$$P\left(o \xrightarrow{d} o\right) = \frac{4}{10} \tag{5.40}$$

Defined as ;

$$P\left(D^i \xrightarrow{w_k} D^j\right) = \frac{e\left(D^i \xrightarrow{x_k} D^j\right)}{\sum_{l=1}^{T} \sum_{m=1}^{A} e\left(D^i \xrightarrow{x_m} D^l\right)} \tag{5.41}$$

The above equation shows that transition from D^i to D^j with $x_k$is equal to count fromD^i to D^j with output x_k divided by total number of counts with D^i as source. Where $e\left(D^i \xrightarrow{x_k} D^j\right)$can be obtained by the following equation;

$$e\left(D^{i} \xrightarrow{X_k} D^{j}\right) = \sum_{s.n+1} P\left(D_{0,n+1} \middle| X_{0,n}\right) * n\left(D^{i} \xrightarrow{X_k} D^{j}, D_{0,n+1}, X_{0,n}\right) \qquad (5.42)$$

The above equation shows that this scheme of picking the valid counts. Considering a singular state sequence for sequence of observations. We get multiple sequence states on an observation sequence, then we weigh number of arrivals by state sequence probability with observation sequence $P\left(S_{0,n+1} \middle| W_{0,n}\right)$.For this we have to interplay between the two equations given above 4.18 and 4.19. Initially we will assume some transition probabilities and then the count is obtained from these probability values. We can also obtain the value of$P\left(S_{0,n+1} \middle| W_{0,n}\right)$ from transition probabilities which is nothing but Viterbi algorithm. Now from the count we obtain new transition probabilities and from the new probability value we obtain the new count. Eventually after sometime the algorithm terminates when we see that there is no appreciable change in the probability values. This algorithm is called Expectation Maximization because we expect a value for the count and then maximizing observation probability sequence through this.

5.8.3.1 Baum-Welch Illustration

Baum Welch comprehends probability values on arcs but not on HMM formation. Illustrating with the following example that consists of (i) two symbols a and b(ii) two states q and r.

Initially we have assumed the transition probabilities and then calculate count from these transition probabilities. Again from this count we will calculate new transition probabilities. This procedure is shown in the table shown below:

String: ababb

$\epsilon \to a$	$a \to b$	$b \to a$	$a \to b$	$b \to b$	$b \to \epsilon$	Path(P)	$q \xrightarrow{a} r$	$r \xrightarrow{b} q$	$q \xrightarrow{a} q$	$q \xrightarrow{b} q$
Q	R	Q	R	Q	Q	0.00077	0.00154	0.00154	0	0.00077
Q	R	Q	Q	Q	Q	0.00442	0.00442	0.00442	0.00442	0.00884
Q	R	Q	R	Q	Q	0.00442	0.00442	0.00442	0.00442	0.00884
Q	R	Q	Q	Q	Q	0.02548	0.0	0.0	0.05096	0.07644
Round Total						0.035	0.01	0.01	0.06	0.095
New Probab -ilities							0.06 (0.01/0.01+0.06 +0.095)	1.0	0.36	0.581

Table 5-5 Baum Welch Algorithm Example

5.8.4.2. Baum Welch Algorithm Computational Complexity

Computational part of Baum Welch Algorithm.

$$E\left(d^i \xrightarrow{x_k} d^j\right) = \frac{1}{P(X_{1,n})}\left[P(D_{1,n+1}, X_{1,n}) * n\left(d^i \xrightarrow{x_k} d^j, D_{1,n+1}, X_{1,n}\right)\right. \tag{5.43}$$

$$P(D_{1,n+1}, X_{1,n}) * n\left(d^i \xrightarrow{x_k} d^j, D_{1,n+1}, X_{1,n}\right) = \sum_{t=1}^{n} P(D_t = d^i, D_{t+1} = d^j, X_t = X_k, D_{1,n+1}, X_{1,n})$$

$$\sum_{t=1}^{n} \alpha_i(t) P(d_i \xrightarrow{x_k} d_i), \beta_i(t+1) \tag{5.44}$$

CHAPTER 6

STATISTICAL VIDEO MODELLING

The basic idea for statistical denoising is to explain the adaptation properties in an expansion of a function into a series of localized basis function. Wavelets finds its usefulness in broad range of applications such as signal and image processing , data compression , numerical analysis and non-parametric statistical estimation

6.1 2D- DWT

Wavelet transform for a video frame deals with its decomposition into a number of detail or wavelet coefficients $\{\psi^{LH}, \psi^{HL}, \psi^{HH}\}$ and one scaling coefficient ϕ^{LL}, forming orthonormal basis $L^2(R^2)$.

M × M image z(t) with given J-scale DWT gets decomposed

$$z(t) = \sum_{i \in Z^2} u_{J,i} \phi_{J,i}^{LL}(t) + \sum_{b \in B} \sum_{i=1}^{J} \sum_{i \in Z^2} w_{j,i}^{b} \psi_{j,i}^{b}(t) \qquad (6.1)$$

Where

$$u_{J,i} = \int x(t) \phi_{j,i}(t) dt \qquad \therefore \text{Coefficient for scaling}$$

And

$$w_{j,i}^{b} = \psi_{j,i}^{b}(t) dt \quad \therefore \text{(i)th wavelet coefficient in j scale with sub-band } \mathcal{B}$$

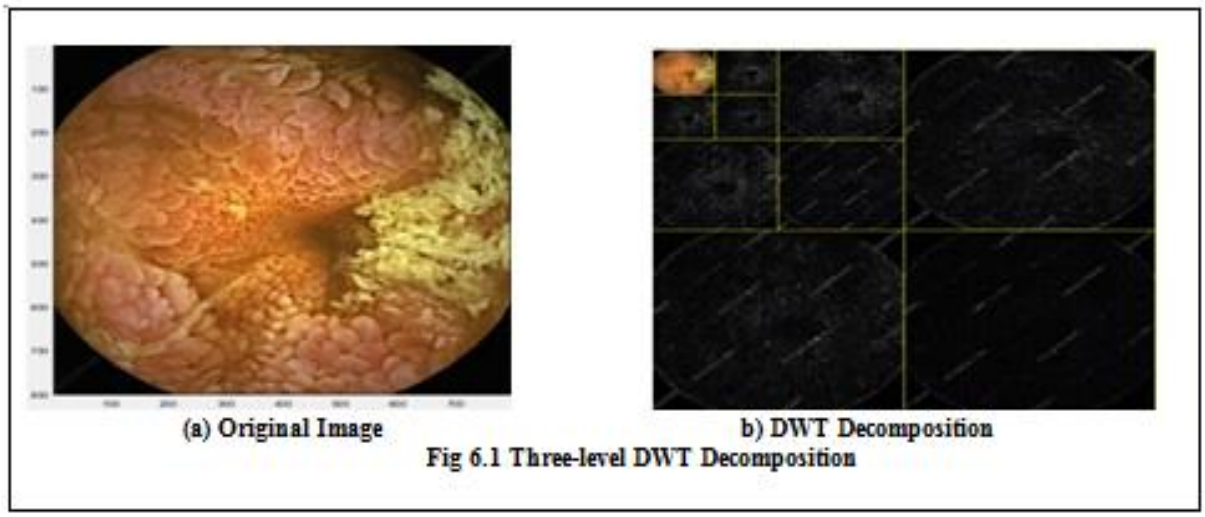
$$\phi_{j,k,i}^{LL}(s,t) = 2^{-\frac{j}{2}} \phi(2^{-j}\ s - k, 2^{-j}\ t - i)$$

$$\psi_{j,k,i}^{LL}(s,t) = 2^{-\frac{j}{2}} \psi^{B}(2^{-j}\ s - k, 2^{-j}\ t - i)$$

$$B \epsilon \mathcal{B}, \mathcal{B}\ \{LH, HL, HH\}$$

$$N_j = N/2_j$$

Wavelets have been utilized in numerous restorative imaging applications. Here 2D-DWT has been used. Image here is decomposed into three level coefficients i.e. LL- subband (low frequency) and several high frequency sub-bands by 2D-DWT. LL-subband contains the most data in concentrated of highest level known as approximated DWT.



(a) Original Image b) DWT Decomposition

Fig 6.1 Three-level DWT Decomposition

6.2 Hidden Markov Model for Video Denoising

When the models are constructed statistically, HMM complex dependencies with captures Non-Gaussian statistics due to the wavelet coefficients, persistence and clustering properties. Here HMM model used Quad tree structure. Wavelet coefficients are linked with a state variable i.e. every wavelet coefficient was described by q(m-dimensional state probability) and σ (m-dimensional standard deviation vector).

$$q = (q_1, q_2, \dots \dots q_m) \tag{6.2}$$

$$\sigma = (\sigma_1, \sigma_2, \dots \dots \sigma_m) \tag{6.3}$$

A multidimensional Gaussian Mixture Model is referred as HMT. Wavelet coefficients are randomly modeled by HMT, with probability density function as a mixture of zero mean Gaussian distribution hidden state for the classification of large and small coefficients.

Where pdf of C

$$f_C(c) = \sum_{n=1}^{N} p_Q(n) f_{c|Q}(c|Q = n) \tag{6.4}$$

$p_Q(n)$ is pmf, and Q is a hidden state random variable. Conditional pmf $f_{c|Q}(c|Q = n)$ is given by following equation

$$f_{c|Q}(c|Q=n)=\frac{1}{\sigma_n\sqrt{2\pi}}\exp\left(-\frac{(b-\mu_n)2}{2\sigma_n^2}\right) \qquad (6.5)$$

Where $\mu_n$and $\sigma_n$are the mean and variance respectively.

HMT used probabilistic tree model to display Markovian dependencies among hidden states to capture inter-scale and intra-scale dependencies present in wavelet coefficients. For decomposition of wavelets into u scale and v sub-band. HMT model has following parameters:
Standard Deviation$=\sigma_{u,v}$, Gaussian mean $=\mu_{u,v}$
Pmf for the root node $Q_i=p_{si}(n)$
Probability matrix for state transition of v sub $-$ band from scale u $-$ 1 to scale u $=A_{u,v}$
The following shows state transition matrix as parent $\rightarrow$ children state to state connection between hidden states:

$$A_{u,v}=\begin{bmatrix} p_{u,v}^{y\rightarrow y} & p_{u,v}^{y\rightarrow z} \\ p_{u,v}^{z\rightarrow y} & p_{u,v}^{z\rightarrow z} \end{bmatrix}$$

where $p_{u,v}^{y\rightarrow y}$or $p_{u,v}^{z\rightarrow z}$ represents wavelet probability to be large or small given the parent is large or small. All these variables are coefficients grouped in θ.

$$\theta=\left[p(Q_i=n),A_{u,v},\mu_{u,v},\sigma_{u,v}\right] \qquad (6.6)$$

Every wavelet here, has different transition state probability. Variances of which leads toward higher complexity in HMT model. It can be reduced by tying within scale method [19].

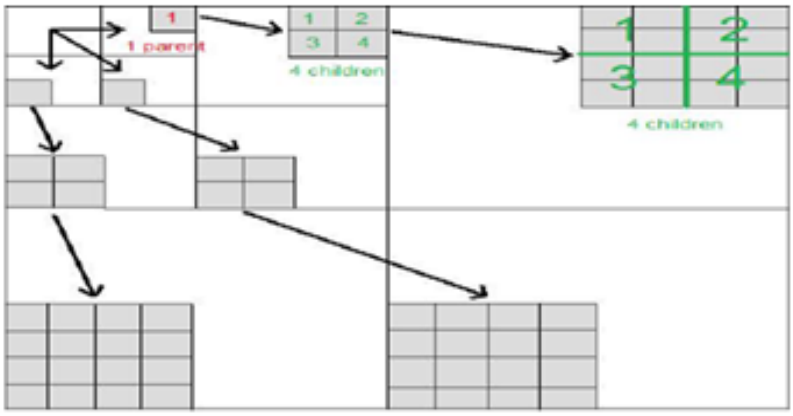


Fig 6.2 Parent-child relationship

CHAPTER 7

Simulation Results

This chapter consists proposed denoising frame work tested through HMT denoising technique with 2D-DWT and 2D-GMM. Maximum likelihood of the data set is found through EM algorithm. Our proposed strategy used the adequacy of DWT and their sub-band hierarchical relationship.

7.1DENOISING TECHNIQUE

HMT model was used to locate a parameter set θ_q.A two state GMM was utilized to start the HMT model. At that point, to get θ_q, the inter-scale dependencies were caught by the Markov-tree and EM-algorithm.
Increase in the signal [20] variance is based on added noise while the other parameters are left unchanged. Noisy observation θ_q was extracted and then noise variance was subtracted from it:

$$(\sigma^{(q)}_{(u,v,m)n})^2 = ((\sigma^{(q')}_{(u,v,m)n})^2 - \left(\theta^{(q)}_{(u,v,m)}\right)^2)_+ \tag{7.1}$$

Where v, u and m represent v sub-band, u scale, n state, mth coefficient and

$$\begin{cases} (g)+= g, & for\ g > 0 \\ (g)+= 0, & for\ g < 0 \end{cases}$$

7.2Model Training via EM Algorithm

EM algorithm is used for model training. The Expectation Maximization algorithm given below, describes the statistical model for hidden state Q and variable C

$$E_{\theta_*}(Q_T(C, Q|C = c) = E_\theta Q_T(C, Q) \tag{7.2}$$

Conditional pmf of state Q and its maximization is given as

$$P(Q = n|C, \theta_q) = \frac{P(Q_i)g(C; 0, \sigma^2_{u,v})}{\sum_{i=0}^{1} P(Q = l)g(C; 0, \sigma^2_{u,l})} \tag{7.3}$$

$$P(Q = n) = \frac{1}{N}\sum_{\square} P(Q = n|C, \theta_q) \tag{7.4}$$

After determining $\theta_q$and state probability via HMT
We got $q = E[q|q', \theta_q]$
Bayes Estimator can be used to obtain clean coefficients.

$$q = \sum_{n} (Q|q, \theta_q) \times \frac{\left(\theta^{(q)}_{(u,v,m)n}\right)^2}{\left(\theta^{(q')}_{(u,v,m)n}\right)^2 + \left(\theta^{(\epsilon)}_{(u,v,m)}\right)^2} q'_{u,v,m} \qquad (7.5)$$

7.3 Inverse Wavelet Transform (IDWT)

In end, IDWT was used to obtain clean coefficients to achieve reconstructed frames of video being tested.

Fig. 7.1 shows proposed implemented technique.

Algorithm for proposed denoising method is summarized as follows:

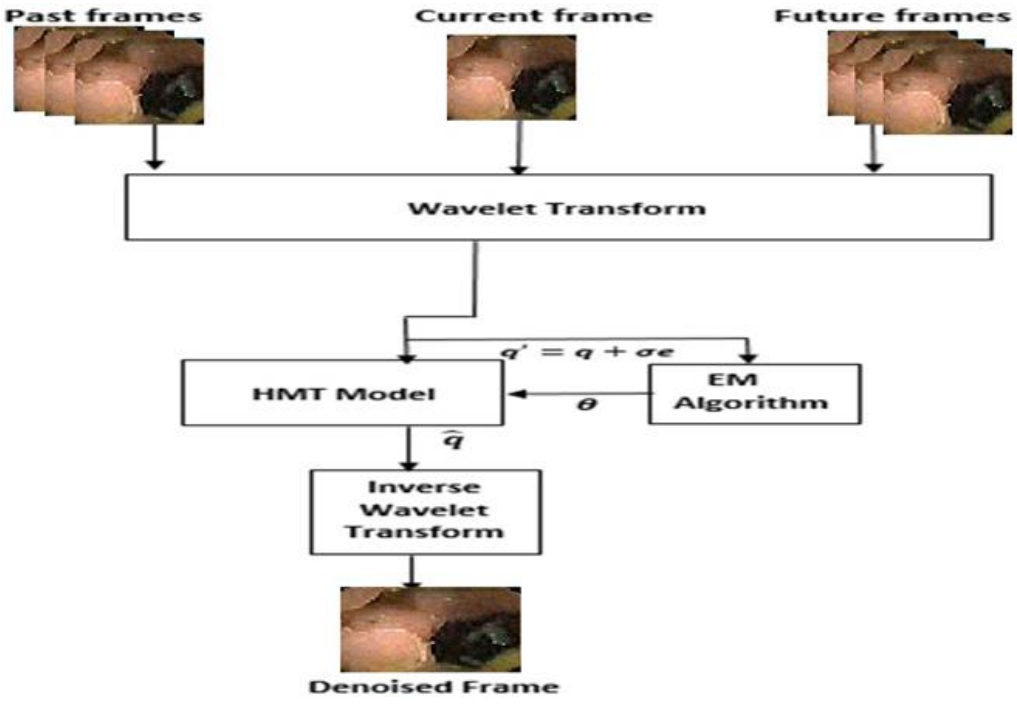


Fig .7.1 Diagram of Proposed Algorithm

7.4 Denoising Algorithm

Adding noise AWGN in each frame of the video sequence

Then applying Daubechies-8 DWT.

Obtaining DWT coefficients.

Estimating GMM parameters.

Training of Hidden Markov Tree model with EM algorithm in connection with tying within scale method.

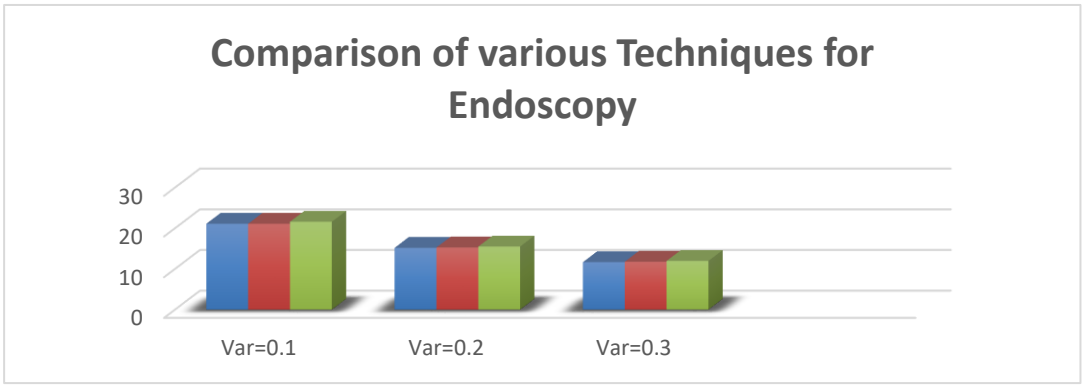
Applying IDWT to get reconstructed frames

7.5 Simulation Results

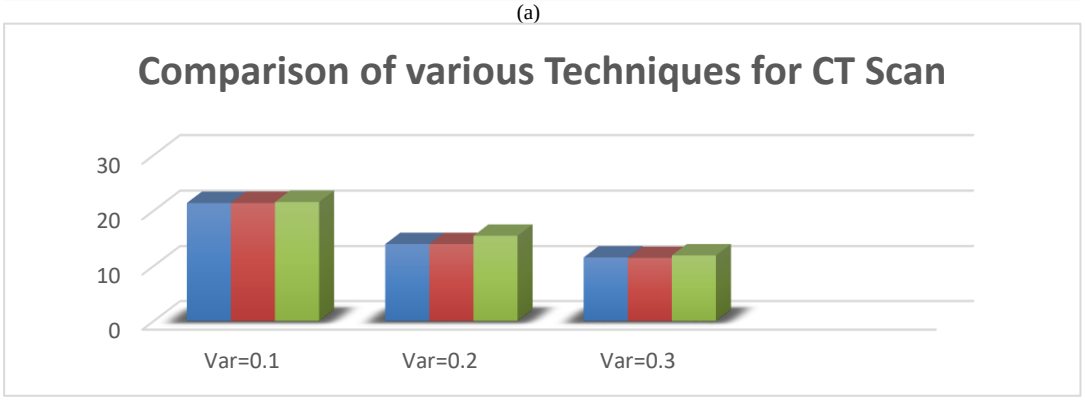
Medical videos were tested i.e. Endoscopy, Ultrasound , Mammogram and CT Scan. Each frame of the sequence was degraded artificially with speckle noise with noise variances i.e.10,20,30. The sequences were tested in grayscale, RGB and proposed algorithm color space. Comparison was performed in terms of PSNR(Table.1).

Table 7.1 PSNR and SSIM values of Denoised Sequences

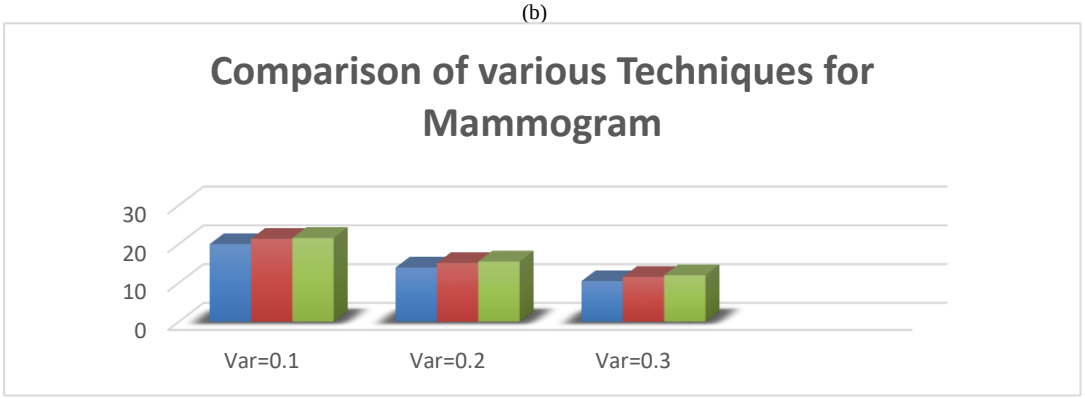
Video Sequences	σ_n	Grey scale		RGB		Proposed algorithm	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Endoscopy	10	21.1167	0.881694	21.1131	0.904999	21.5982	0.884287
	20	15.1980	0.822676	15.2762	0.894293	15.5062	0.828196
	30	11.6759	0.855640	11.7776	0.909898	11.9761	0.761609
CT Scan	10	21.3809	0.866745	21.3847	0.866680	21.5603	0.852319
	20	13.9769	0.846671	13.9789	0.867870	15.4741	0.854179
	30	11.5437	0.826184	11.4503	0.859014	11.9199	0.837778
Mammogram	10	20.0024	0.836505	21.3705	0.867967	21.6039	0.856158
	20	13.9792	0.830533	15.1374	0.876485	15.5392	0.846681
	30	10.4566	0.831411	11.5517	0.853997	11.9808	0.813299
Ultrasound	10	21.1587	0.845638	21.2203	0.905396	21.4992	0.883864
	20	13.9793	0.877209	15.3598	0.885453	15.5102	0.864535
	30	11.6582	0.848289	11.8212	0.903502	11.9896	0.830726



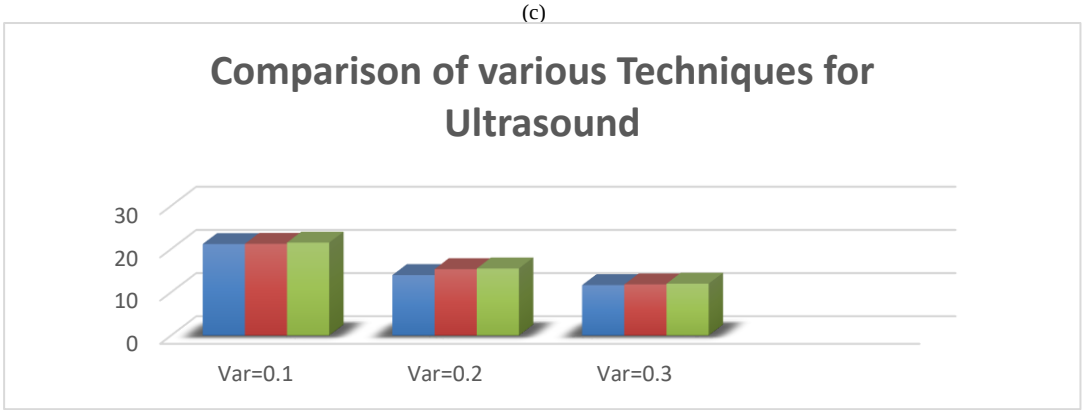
(a)



(b)

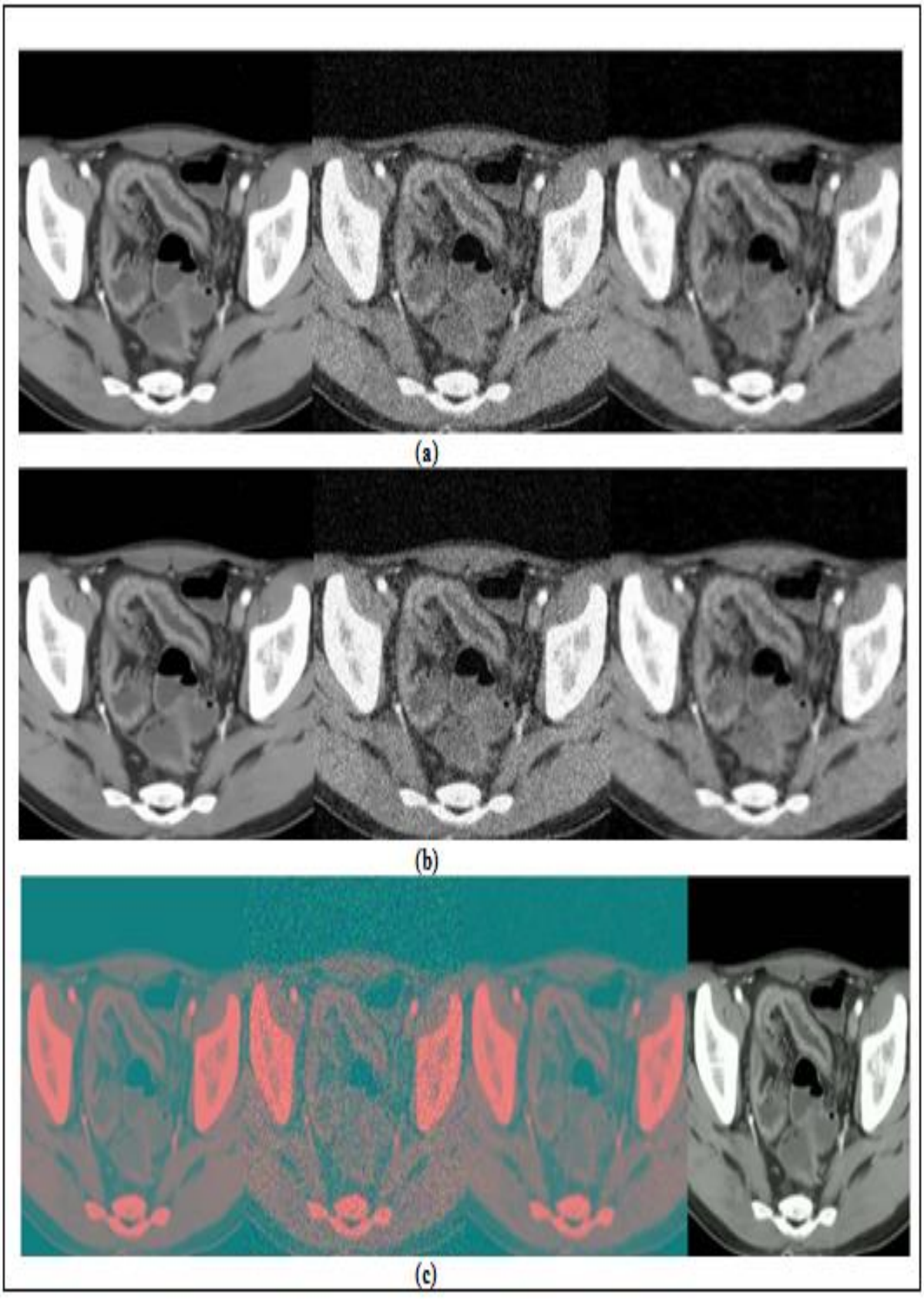


(c)



(d)

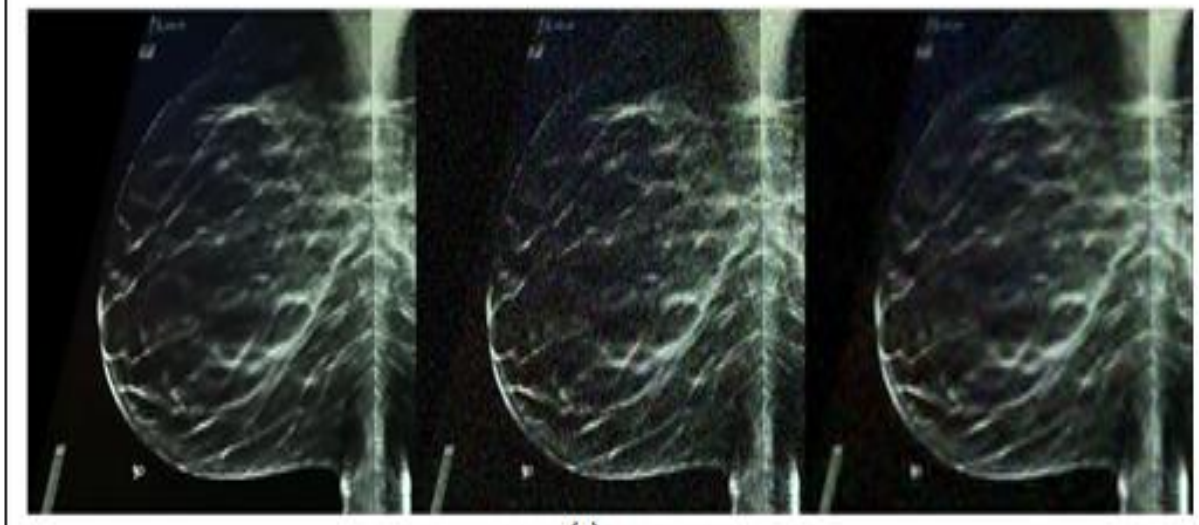
Figure 4. GRAPHICAL COMPARISON OF PSNR OF VARIOUS TECHNIQUES


(a)
(b)
(c)

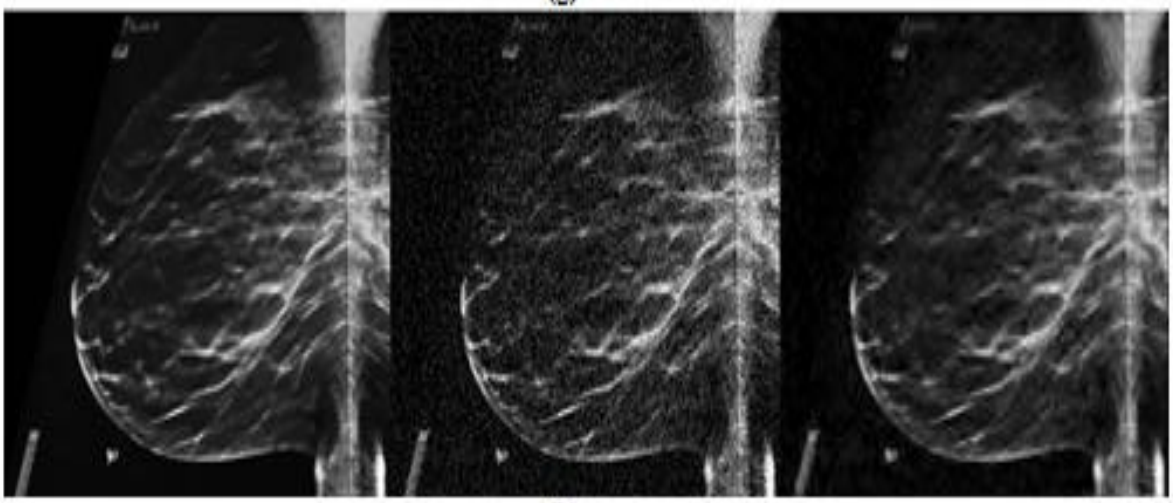
(d)

(e)

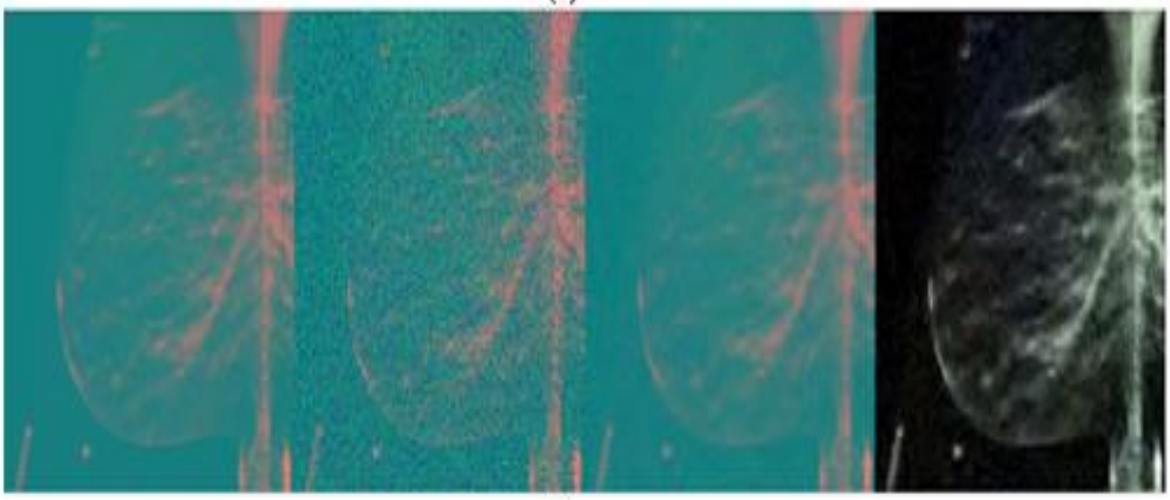
(f)



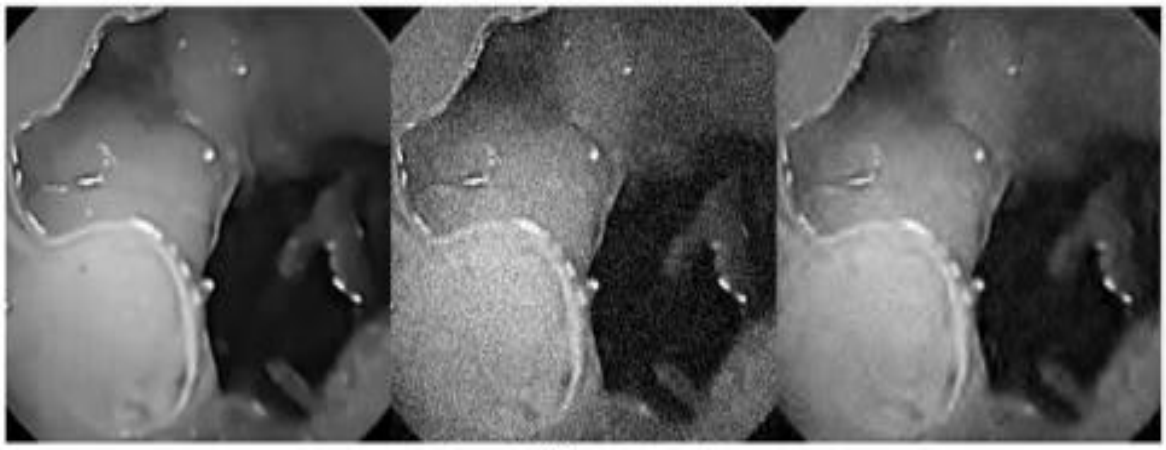
(g)



(h)



(i)



(i)

(k)

(l)

QUALITATIVE COMPARISON OF PSNR OF DENOISING OF DIFFERENT VIDEO SEQUENCES USING DIFFERENT TECHNIQUES WITH UNIFORM GAUSSIAN NOISE (a,d,g,j) GRAY SCALE (b,e,h,k) RGB (c,f,i,l) PROPOSED ALGORITHM

CHAPTER 8

Conclusion

Wavelet-based color video denoising was discussed in this thesis was based on HMT model with EM algorithm for the despeckling of video frames scale videos and colored videos in YCbCr color space based on 2D-DWT and 2D-GMM. The primary properties of the compression , wavelet transform locality and multi-resolution paved way for new approaches towards statistical signal processing.

To accommodate Non-Gaussian nature of wavelet coefficients, Mixture densities have been incorporated and statistical dependencies between coefficients were captured by using probabilistic graphs/Tree.HMT model was trained by EM algorithm on wavelet coefficients to catch the statistical dependencies present in them.

Results revealed that proposed denoising method transcends the pre-existing techniques in comparison with gray and RGB scale. both in terms of quantitative and qualitative analysis. This scheme showed improvement in reducing noise , preservation of edges with enhanced visual quality.

8.1 Future Work

Proposed denoising scheme can be further extended for the analysis of noises of different variances with other transforms for all color spaces. Moreover, it can be used in other transform domains like Bandelet, Contourlet, Rigdelet and Curvelet in combination with other filters i.e. Bilateral filter.

Finally, this technique can be further modified with other techniques for the achieving improved performance and with reduced computational complexity and low latency.

Bibliography

1. Sharma, A., and Singh, J., “Image Denoising Using Spatial Domain Filters: A Quantitative Study”, Proceedings of 6th IEEE International Congress on Image and Signal.
2. Narasimha, C., and Rao, N.A., “Spatial Domain Filter for Medical Image Enhancement”, Proceedings of IEEE International Conference on Signal Processing and Communication Engineering Systems,Guntur, India,January, 2015.
3. Wang, B., Xiong, Z., and Zhang, D., “Nonlocal Image Denoising via Collaborative Spatial-domain LMMS Estimation”, IEEE International Conference on Image Processing, Paris, France, October, 2014.
4. Lee, J.S., “Digital Image Enhancement and Noise Filtering by Use of Local Statistics”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 2,No. 2, pp.165-168, 1980.
5. Li, X., Shen, H., and Zhang, L., “Sparse-Based Reconstruction of Missing Information in Remote Sensing Images from Spectral/Temporal Complementary Information”, ISPRS Journal of Photogrammetry and Remote Sensing, Volume 106, pp. 1-15, 2015.
6. Manjón JV, Coupé P, Buades A, Collins DL, Robles M. New methods for MRI denoising based on sparseness and self-similarity. Med Image Anal.2012;16(1):18-27.
7. Aksam Iftikhar M, Jalil A, Rathore S, Hussain M. Robust brain MRI denoising and segmentation using enhanced non-local means algorithm. Int J Imaging Syst Technol. 2014;24(1):52-66.
8. Liu RW, Shi L, Huang W, Xu J, Yu SCH, Wang D. Generalized total variation-based MRI Rician denoising model with spatially adaptive regularization parameters. Magn Reson Imaging. 2014;32(6):702-720.
9. Ashish Khare, Uma Shanker Tiwary, "Daubechies Complex Wavelet Transform Based Technique for Denoising of Medical images", Article in International Journal of Image and Graphics, October 2007.
10. C.P. Loizou and C.S. Pattichis, “Despeckle filtering algorithms and Software for Ultrasound Imaging”, Synthesis Lectures on Algorithms and Software for Engineering, Ed. Morgan & Claypool Publishers, San Rafael, CA, USA, 2008.

11. Varun P. Gopi, P. Palanisamy,“Capsule endoscopic image denoising based on double density dual tree complex wavelet transform”, Article in International Journal of Imaging and Robotics. January 2012.
12. Christos P. Loizou1, Takis Kasparis, Pavlos Christodoulides, Charoula Theofanous, Marios Pantziaris, Efthyvoulos Kyriakou4, Constandinos S. Pattichis,”Despeckle Filtering in Ultrasound Video of the Common Carotid Artery”, Proceedings of the 2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE), Larnaca, Cyprus, 11-13 November 2012.
13. H.Rabbani, S. Gazor ,"Video denoising in three-dimensional complex wavelet domain using a doubly stochastic modelling,” IET Image Processing , Vol 6, December 2012.
14. Manoj Kumar, Manoj Diwakar , “CT image denoising using locally adaptive shrinkage rule in tetrolet domain”, Babasaheb Bhimrao Ambedkar University, Lucknow, India, 17 March 2016.
15. Sugandha Agarwal, O.P.Singh, Deepakk Nagaria, "Analysis and Comparison of Wavelet Transforms For Denoising MRI Image”, May 10, 2017
16. Azka Maqsood, ImranTouqir, Adil Masood Siddique, MahamHaider, "Wavelet Based Video Denoising using Probabilistic Models", Mehran University Research Journal of Engineering &Technology,Vol. 38, No. 1, 17-30, January 2019.
17. Michele Claus, Jan van Gemert “ViDeNN: Deep Blind Video Denoising”, Computer Science Published in CVPR Workshops 2019.
18. Rajeshwar Dass,"Reduction of Ultrasound Images Using BFO Cascaded with Wiener Filter and Discrete Wavelet Transform in Homomorphic Region”, International Conference on Computational Intelligence and Data Science, 2018.
19. Malfait, M., and Roose, D., “Wavelet-Based Image Denoising Using a Markov Random Field a Prior Model”, IEEE Transactions on Image Processing, Volume 6, No. 4, pp. 549-565, 1997. Crouse,
20. M.S., Nowak, R.D., and Baraniuk, R.G., “Wavelet-Based Statistical Signal Processing using Hidden Markov Models”, IEEE Transactions on Signal Processing, Volume 46, No. 4, pp. 886-902, 1998.

21. Golshan HM, Hasanzadeh RP, Yousefzadeh SC. An MRI denoising method using image data redundancy and local SNR estimation. Magn Reson Imaging.2013;31(7):1206-1217.