ISSN:0975-9646

Allam Mohan et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (4) , 2012,4812-4817

A New Blind Image Watermarking using Hermite Spline Approach

Allam Mohan[1], Kampalli Ramu[2], Salina Adinarayana[1]

[1] *Shri Vishnu Engineering College for Women, Bhimavaram , AndhraPradesh, India*
[2] *Adams Engineering College, Paloncha, AndhraPradesh, India*

Abstract— **This paper presents a robust image watermarking scheme based on a sample spline approach. While we consider the human visual system in our watermarking algorithm, we use the low frequency components of image blocks for data hiding to obtain high robustness against attacks. We use four samples of the approximation coefficients of the image blocks to construct a spline curve in the 2-D space. The slopes of this spline curve, which is invariant to the gain factor, is employed for watermarking purpose. We embed the watermarking code by constructing a spline curve according to message bits. To design a maximum likelihood decoder, we compute the distribution of the slope of the embedding spline curve for Gaussian samples. The performance of the proposed technique is analytically investigated and Experimental results confirm the validity of our model and its high robustness against common attacks in comparison with similar watermarking techniques that are invariant to the gain attack.**

Keywords: Image watermarking, Maximum Likelihood detector, Hermite Spline, Bresenham's Line, Fibonacci LSB, Bessel interpolator, gain attack.

I. INTRODUCTION

Digital watermarking embeds information within a digital work as a part of the media. Watermarking techniques falls into three categories of robust, semi fragile and fragile methods according to their specific applications [1]–[3]. Robust watermarking mainly serves for identification purposes while the fragile and semi fragile watermarking are usually employed in authentication applications. Since a good watermarking scheme should always be able to deal with some kinds of attacks, studies in the watermarking research area mostly target robust watermarking problems.

Several robust watermarking techniques have been proposed so far. Cox et al. [4] have proposed an additive watermarking approach based on spread spectrum concept which remains highly robust against noise and cropping attacks. Based on this observation that boosting the watermarking power increases the barrier against attacks, most of the effective watermarking schemes try to match the characteristics of the watermark to those of the image asset. Multiplicative watermarking, as an example, has been introduced in [5] and has been widely studied later on using local optimum decoders in multi resolution transform domains such as wavelet and contour let domains [6]–[10]. Besides, a universal optimal detector for scaling based watermarking schemes is presented in [11]. These schemes are highly robust against noise and compression attacks. To satisfy robustness against geometric attacks and reduce the watermark synchronization problem, Thus, no approach has been presented so far that both proposes an optimal decoder and remains invariant to the gain attack.

In this paper, we proposed a novel gain invariant watermarking scheme based on a sample spline scheme. Embedding the watermark bits into the approximation coefficients of the image blocks makes the algorithm highly robust against noise and compression attacks. Any possible selection of four approximation coefficients, that may be selected using a secret key, constructs a spline curve whose slope at control points is considered for data hiding. We embed the watermark bits by constructing a spline curve with the tangents at control points (slope). In this way, the slope of the spline curve carries the watermark information while the distortion imposed to its constructive samples is minimal. Since our embedding process is linear, it can be denoted by multiplication of specific embedding matrices. To implement the maximum likelihood (ML) detector for data extraction, we should calculate the distribution of the slope of the spline curve. To this aim, we consider the fact that the approximation coefficients of most of the image blocks can be well-modelled by Gaussian distribution [9]. The performance of our watermarking method is analytically investigated and verified via simulation on artificial signal. Several experiments on sample images confirm the effectiveness of the proposed scheme in resisting against common noise attacks.

The rest of the paper is organized as follows. In Section II, we describe the model of the system. The watermark embedding and decoding process are introduced in Section III. Section IV analyzes and evaluates the performance of the proposed scheme. Experimental results are demonstrated in Section V, and Section VI concludes the paper.

II. SYSTEM MODELING

In this section, we first introduce the model considered for our watermarking algorithm. To this aim, we calculate the distribution of the watermarking variable. We assume to have four samples of an independently and identically distributed (i.i.d) Gaussian random variable as the host signal. We show this signal as u = [u1 , u2 , u3 , u4] with the Gaussian distribution of $N(0, \sigma^2 u)$. These four samples form two points P1 = [u1 , u2] and P2 = [u3 , u4] in the 2-D space. We employ Dp_k, the slope of the spline at first control point and Dp_{k+1}, the slope of the spline at second control point as our watermarking variables.Fig.1 illustrates these two points as well as the curve derivatives ('slopes') at these points . We can write the procedure as

- Algorithm will generate a cubic curve.

Fig. 1.Hermite Specification Control points P_0 ,P_1 and Slopes $P_{0'}$, $P_{1'}$

Let the parametric curve be P(u) = au^3 + bu^2 + cu + d. where u is the parameter that ranges from 0 to 1.A defined Hermite curve has a defined set of coefficients a, b, c, d. Substituting a value *u* into the equation gives a point on the Hermite curve. Substituting many values of *u* from 0 to 1 will trace out the curve. Given the two points and two slopes, p_0, p_1, p_0' and p_1', our objective is to find the coefficients a, b, c, d.

$$P(u) = au^3 + bu^2 + \mathrm{cu} + \mathrm{d}$$

$$P(u) = [u^3\ u^2\ \mathrm{u}\ 1]\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

Derivative of $P(u)$ is,

$$P'(u) = [u^3\ u^2\ \mathrm{u}\ 1]\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

But,

$$P(0) = p_0 = a{\times}0^3 + b{\times}0^2 + c{\times}0 + \mathrm{d} \times 1$$
$$P(1) = p_1 = a{\times}1^3 + b{\times}1^2 + c{\times}1 + \mathrm{d} \times 1$$
$$P'(0) = p_0{}' = a{\times}3{\times}0^2 + b{\times}2{\times}0 + \mathrm{c} \times 1 + \mathrm{d} \times 0$$
$$P'(1) = p_1{}' = a{\times}3{\times}1^2 + b{\times}2{\times}1 + \mathrm{c} \times 1 + \mathrm{d} \times 0$$

Therefore, in matrix form,

$$\begin{bmatrix} p_0 \\ p_1 \\ p_0{}' \\ p_1{}' \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

Solve for a,b,c,d by using matrix inverse:

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_0{}' \\ p_1{}' \end{bmatrix}$$

We get:

a = $2p_0$ - $2p_1$ + $p_0{}'$ + $p_1{}'$
b = -$3p_0$ + $3p_1$ -2 $p_0{}'$ - $p_1{}'$
$c = p_0'$
$d = p_0$

Next, Substitute back to equation

$$P(u) = au^3 + bu^2 + \mathrm{cu} + \mathrm{d}$$

We get:

$$P(u) = (2p_0 - 2p_1 + p_0' + p_1')u^3 + (-3p_0 + 3p_1 - 2p_0' - p_1')u^2 + p_0'\mathrm{u}$$

Rearranging.

$$P(u) = p_0(2u^3 - 3u^2 + 1) + p_1(-2u^3 + 3u^2) + p_0'(u^3 - 2u^2 + u) + p_1{}'p_1(u^3 - u)$$

Now, to embed the *M*-ary watermark code, we use this equation to generate Hermite spline curve given control points and slopes shown in Fig. 1, depending on the watermark code. In this way, we obtain the watermarked signal.

To extract the hidden bits, an optimum decoder is implemented using M-Hypothesis test as follows. We take the received watermarked signal which consists of the Hermite Spline with two control points and some points on spline curve and calculate the slopes of the Hermite spline curve at each control point. Here the slopes are nothing but the watermarked bits which we have sent it with Host image signal.

III. PROPOSED METHOD

In this section, we introduce our blind watermarking scheme. As discussed in the previous section, we assume the host signal as a four-sample i.i.d. Gaussian random signal. In practical applications, these four samples can come from approximation coefficients of the image blocks which satisfy our i.i.d. Gaussian assumption according to Kolmogorov Smirnov test results.

Figure 2. Shows a model of watermarking that allows watermark pattern **Wa** to be dependent on original cover work **Co**. we can get the final watermarked image **Cw** from the Watermark embedder. **Cwn** is the watermarked image with some noise mixing, which we are providing to the Watermark detector. Finally we can get the Watermark as an Output message from detector.

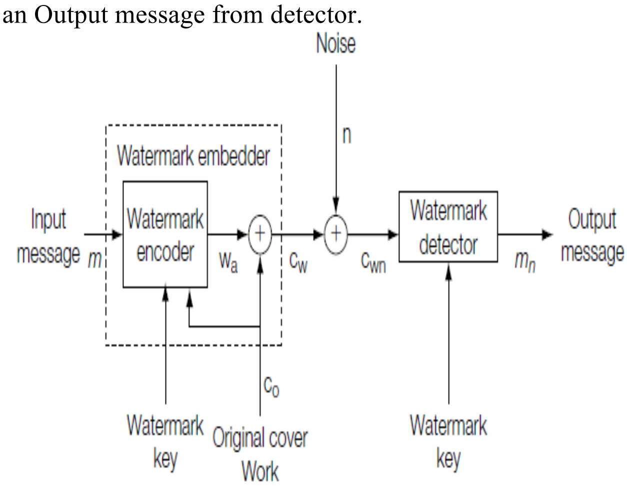

Fig. 2 watermarking as communications with side information at the transmitter.

A. Watermark embedding

We use Hermite cubic interpolation for embedding watermark with the help of four samples as control points and Watermark embedding bits as Slopes at each control point. The interpolator used to construct the curve generally has a parametric representation, i.e. the curve is a two dimensional function of an underlying parameters s,

$x = f(s)$ (1)

Which passes through the collection of sample points *{Xi}*,
$x_i = f(s_i)$ (2)
This notation allows for easy extension to three dimensions [12]. The parametric notation also gives an interpolator that is isotropic, i.e. invariant with respect to orientation. The parameter s is usually related to the arc length along the curve. However, using the chord length

$$s_{i+1} = s_i + d_i$$

Where

$$d_i = |d_i| = |x_{i+1} - x_i|$$

with s varying linearly between control points, will give equally good results [13]. A further simplification, $s_i = i$, can be made if the point spacing is fairly uniform.

The interpolation requirement precludes the use of some functions such as quadratic B-splines [14] which are otherwise suitable curve generators. One simple class of functions that both interpolates and can be made sufficiently smooth are the cubic splines and sub-splines. These functions are piecewise cubic, i.e. on any interval (s_i, s_{i+1}) the function is a cubic

$$f_i(s) = a_{3i}s^3 + a_{2i}s^2 + a_{1i}s + a_{0i}$$

For sufficient smoothness, we require that the curve be first derivative (C^1) continuous. Such curves are variously known as sub-splines or Hermite cubics. They are completely defined by the control points $\{x_i\}$ and the curve derivatives ('slopes') at these points $\{\bar{x}_i\}$ (Fig. 3). Each cubic segment can also be represented as a linear combination of four basis or 'blending' [15] functions weighted by the end points and the end-point derivatives

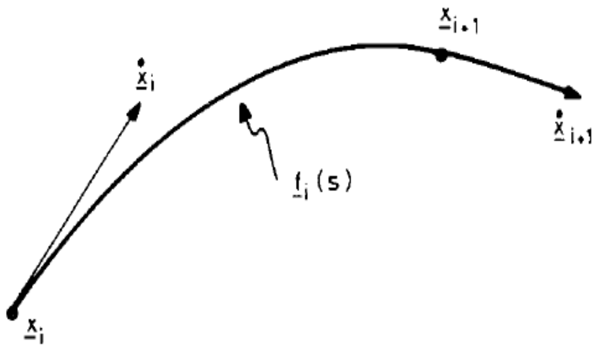

Fig. 3 Cubic segment defined by its end points and end-point derivatives

$$f(s_i) = x_i\phi_0(t) + \bar{x}_i\phi_1(t) - \bar{x}_{i+1}\phi_1(1-t) + x_{i+1}\phi_1(1-t), t = \frac{s-s_i}{d_i} \in [0,1]$$

These basis functions, the Hermite Cubic basis functions, are

$$\phi_0(t) = 2t^3 - 3t^2 + 1$$
$$\phi_1(t) = t(1-t)^2$$

And are shown in Fig. 3.

Since the sample points $\{x_i\}$ are given; only the slopes $\bar{x}_i$ need to be determined in order to specify completely the interpolator. The slope values $\bar{x}_i$ are the true derivative values of the interpolator f(s) , but can only be an estimate of the derivatives of the original curve. The overall Hermite cubic interpolate function f(s) is thus composed of piecewise cubic segments h(s) with first-derivative continuity across interval boundaries.
After finding the points on the spline curve, we embed these points in the pixel positions of a line segment (line joining control points) using A Fibonacci LSB Data Hiding Technique. We can find pixels along a line segment by joining the two control points with the help of Bresenham's Line Drawing Algorithm. Assume $y = mx + c$ represents the real variable equation of a line which is to be plotted using a grid of pixels where the two points (x_1, y_1) and (x_2, y_2) have integer coordinates and represent two known points on the line which are to be connected by drawing a simulated line segment (or segments) which connects them. In figure 4 below these two points are respectively the lower-left and upper-right corners of the pixel grid.

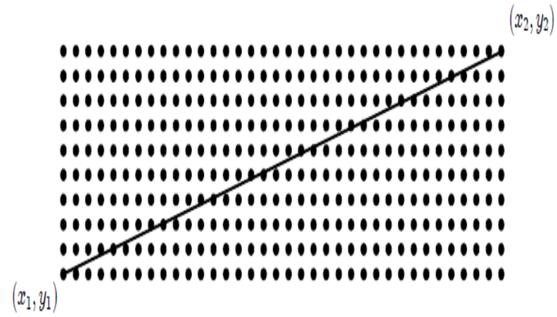

Fig. 4 A true line lay out across a grid of pixels

A Fibonacci LSB is a generalization of the classical Least Significant Bit (LSB) and is one of the simplest technique in digital watermarking is in spatial domain using the two dimensional array of pixels in the container image to hold hidden data. The human eyes are not very attuned to small variance in colour and therefore processing of small difference in the LSB will not noticeable Fibonacci LSB embedding scheme is as follows,
Let us define I (x, y) the cover image, and w (x, y) the data to be embedded. In our experiment they have the same size. The proposed embedding scheme can be summarized as follows:

1. The cover image I (x, y) is decomposed into bit planes I p (x, y) = Fp { I (x, y)} by using the Fibonacci p-decomposition computed using the specified p-sequence.
2. The selected plane is considered. For each bit the fulfilment of Zeckendorf condition is checked. If it is verified, the mark is inserted otherwise the following is considered. The simplest method used in our experiment is the substitution of the selected bit value with the corresponding watermark bit. An improved version is based on the following additive scheme:
   I p (x, y) = I p (x, y) + α w (x, y)

3. Once the whole mark has been inserted, the image is reconstructed from the bit planes. The watermarked image is therefore recomposed With respect to this basic scheme; several modifications have been tested to increase the robustness of the system. Among these, we cite the use of a block-wise embedding scheme in which each block has a different embedding strength value according to some HVS-based features. We are investigating the contrast visibility factor to drive such a scheme.

B. Watermark decoding

To extract the hidden bits, we need to determine the slope the Hermite spline by cubic interpolator. To generate the cubic interpolator, the slopes $\{\overline{x_i}\}$ must be determined. One common way to specify the slopes is to require second derivative (C^2) continuity. The resulting system of tri-diagonal equations can then be solved using various iterative or direct methods [16, 17], with the resulting interpolator being known as the full cubic spline. However, this calculation requires all of the points on the curve to be known, and it is thus not sufficiently local to be used for real-time reconstruction.

We take the received watermarked signal and key for generating the four samples to get two control points. Now we use these points for finding slope of the line which connects them. Once we have the slope of the line, we can generate the line points with the Bresenhams Line drawing Algorithm .Now we can get the spline curve points stored in the points along the line with LSB Algorithm .With these Hermite Spline curve points and control points we can calculate the slopes of the Hermite spline curve at each control point. Here the slopes are nothing but the watermarked bits which we have sent it with Host image signal.

The slopes can also be adjusted interactively. Methods such as Bezier curves [4J have been designed to do this naturally by specifying additional control points. This is not a satisfactory approach if the curve generation is to proceed automatically from a set of sample points without operator intervention. What is required instead is a method to estimate the individual slopes using only a few of the neighbouring points.

The simplest method for determining the slope locally is to use a parabola through a sample point and its left and right neighbours to determine the slope at the point (modifications can be made for the end points of the curve).

The parabolic equation

$$g(s)= \frac{x_i}{2}(s-s_i)^2 + x_i(s-s_i) + x_i$$

With

$$g(s_{i-1}) = x_{i-1} \quad \text{and} \quad g(s_{i+1}) = x_{i+1}$$

Can be solved to yield

$$x_i = \frac{d_{i-1}m_{i,i+1} + d_i\, m_{i-1,i}}{d_{i-1} + d_i}$$

The value m_{ij} j is the divided difference of the curve (Fig. 5),

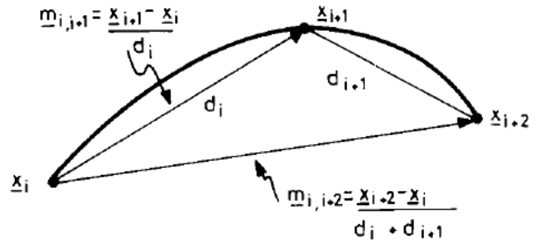

Fig. 5: Divided differences of a curve used to calculate the slopes and are defined as the slope of a line connecting two sample points:

$$F[s_i, s_j] \triangleq \frac{x_i - x_j}{s_i - s_j} \triangleq m_{i,i} \text{ (for short)}$$

The interpolator resulting from this parabolic fit, known as the Bessel interpolator, is quite simple, but does not smooth as well as methods that use more of the neighbouring points. It is, in effect, a four-point interpolator, since the points x_{i-1}, x_i , x_{i+1} and x_{i-2} are needed to define the two end slopes x_i and x_{i+1} for the segment $f_i(s)$.

Higher-order polynomials (such as quadratics) can be fitted to determine the slope, but the exact solution rapidly becomes complex. An assumption of near-uniform spacing can be used to obtain some simple results which take the form of a SUlTI of divided differences weighted by fixed rational numbers (see Appendix A).

IV. Experimental results

In Fibonacci LSB [18] the value of α represents the strength of the watermark. The greater is α, more robust is the watermarking system. Unfortunately, it results also in images in which the user can perceive the artefacts. The same considerations are also valid for the bit plane selected for the embedding. In Figure 6, the original image and different versions of the watermarked image are shown. The upper right one has been obtained by using the classical LSB watermarking scheme. The lower left and right ones are obtained using the proposed method. As can be noticed, the impact of the watermark in the latter image is much stronger than in the third one.

To test the effectiveness of the detection scheme, the detector response to the watermarked image ''Lena'' to 500 randomly generated marks has been considered. As can be noticed in Figure 6, the correlation peak corresponds to the true mark presentation.

In Fig.8: we can see the possible Four Basis Functions for Hermite splines.

Fig. 6: from left to right, up to down. Original image, classical LSB watermarked image, LSB Fibonacci watermarking, first significant bit watermarked image.

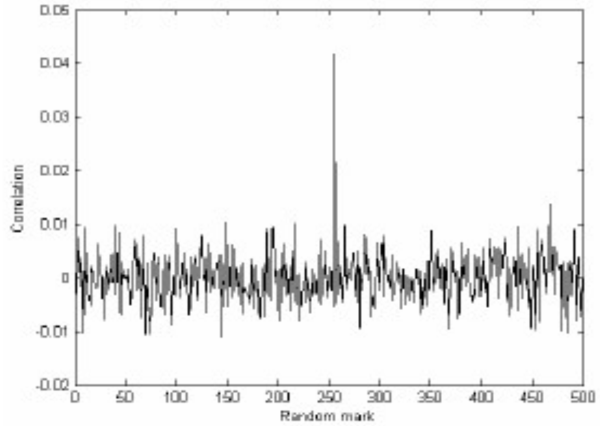

Fig. 7: detector response to the watermarked image 'Lena'' to 500 randomly generated watermarks. Only watermark number 250 matches the embedded one.

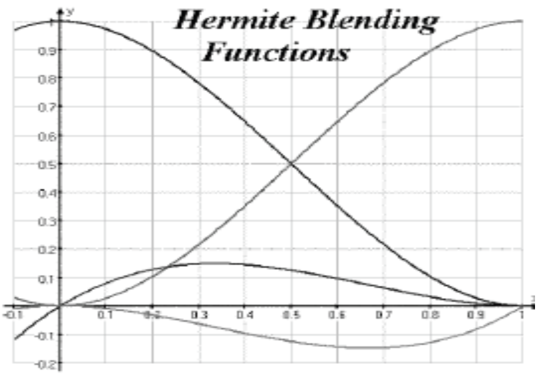

Fig. 8 Four Basis Functions for Hermite splines.

V. CONCLUSION

In this paper, we presented a novel blind watermarking approach with the optimal decoder. Watermark embedding is performed by multiplication of M specific matrices to the vector of samples of size four. These matrices construct a Hermite spline curve with the vector of samples and slopes depending on the message symbol. Assuming the host samples to be i.i.d Gaussian, which is often valid for approximation coefficients of image blocks [15], we obtained a closed form PDF of noisy watermarked samples. Having this distribution function, we designed an optimum ML decoder. We analytically studied and verified the error probability of the proposed decoder in a noisy environment. The proposed algorithm is applied to image signals by using four approximation coefficients of the image blocks which may be selected according to a secret key. In addition to be invariant to the volumetric distortions, several simulations showed that the proposed algorithm is highly robust against common watermarking attacks such as AWGN, compression, and filtering. However, the algorithm is sensitive to the collusion attack. The future work may be proposing a solution to this problem in order to make the algorithm robust to this attack as well.

VI .APPENDIX

This appendix derives the slope estimate obtained by using a quadratic polynomial fit. An assumption of quasi-uniform spacing is made in order to simplify the solution. The quadratic polynomial

$$g_i(s) = a(s - s_i)^4 + b(s - s_i)^3 + c(s - s_i)^2 + \bar{x}_i(s - s_i) + \bar{x}_i$$

is made to pass through the four neighbouring points

$$g_i(s_j) = x_i\ , j = i - 2, i - 1, i + 1\ , +2$$

In matrix notation, we have

$$\begin{bmatrix} -(d_{i-2} + d_{i-1})^3 & (d_{i-2} + d_{i-1})^2 & -(d_{i-2} + d_{i-1}) & 1 \\ -d^3{}_{i-1} & d^2{}_{i-1} & -d_{i-1} & 1 \\ d^3{}_i & d^2{}_i & d_i & 1 \\ (d_i + d_{i+1})^3 & (d_i + d_{i+1})^2 & (d_i + d_{i+1}) & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ \bar{x}_i \end{bmatrix} = \begin{bmatrix} m_{i-2,i} \\ m_{i-1,i} \\ m_{i,i+1}{}' \\ m_{i,i+2} \end{bmatrix}$$

This general matrix inversion must be performed for each point if a non-uniform grid is assumed. However, if a uniform grid $s_i = i$ (and hence $d_i = 1$) is assumed for the left-hand side, a simplified matrix results

$$\begin{bmatrix} -8 & 4 & -2 & 1 \\ -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 \\ 8 & 4 & 2 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ \bar{x}_i \end{bmatrix} = \begin{bmatrix} m_{i-2,i} \\ m_{i-1,i} \\ m_{i,i+1}{}' \\ m_{i,i+2} \end{bmatrix}$$

This can be solved for

$$x_i = \frac{2}{3}\{m_{i-1,i} + m_{i,i+1}\} - \frac{1}{6}\{m_{i-2,i} + m_{i,i+2}\}$$

ACKNOWLEDGMENT

We would like to thank all our colleagues who encouraged us to fulfill this work. This work can't be completed without the support of our families.

REFERENCES

[1] J. Seitz, Digital Watermarking For Digital Media. Arlington, VA, USA: Information Resources Press, 2005.

[2] C.-S. Lu, Multimedia Security: Steganography and Digital Watermarking Techniques for Protection of Intellectual Property. Hershey, PA, USA: IGI Publishing, 2004.

[3] G. Langelaar, I. Setyawan, and R. Lagendijk, "Watermarking digital image and video data. a state-of-the-art overview," IEEE Signal Process.Mag., vol. 17, no. 5, pp. 20 –46, Sep. 2000.

[4] I. Cox, J. Kilian, F. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," IEEE Trans. Image Process., vol. 6,no. 12, pp. 1673 –1687, Dec. 1997.

[5] M. Barni, F. Bartolini, A. De Rosa, and A. Piva, "A new decoder for the optimum recovery of nonadditive watermarks," IEEE Trans. Image Process., vol. 10, no. 5, pp. 755 –766, May. 2001.

[6] "Optimum decoding and detection of multiplicative watermarks," IEEE Trans. Signal Process., vol. 51, no. 4, pp. 1118 – 1123, Apr. 2003.

[7] Q. Cheng and T. Huang, "Robust optimum detection of transform domain multiplicative watermarks," IEEE Trans. Signal Process., vol. 51,no. 4, pp. 906 – 924, Apr. 2003.

[8] J. Wang, G. Liu, Y. Dai, J. Sun, Z. Wang, and S. Lian, "Locally optimum detection for barni's multiplicative watermarking in dwt domain," Signal Process. vol. 88, no. 1, pp. 117–130, 2008.

[9] M. A. Akhaee, S. M. E. Sahraeian, B. Sankur, and F. Marvasti, "Robust scaling-based image watermarking using maximum-likelihood decoder with optimum strength factor," IEEE Trans. Multimedia, vol. 11, no. 5, pp. 822 –833, Aug. 2009.

[10] M. A. Akhaee, S. M. E. Sahraeian, and F. Marvasti, "Contourlet-based image watermarking using optimum detector in a noisy environment," IEEE Trans. Image Process., vol. 19, no. 4, pp. 967 – 980, Apr. 2010.

[11] S. M. E. Sahraeian, M. A. Akhaee, and F. Marvasti, "Distribution Independent blind watermarking," in Proc. IEEE Int. Conf. Image Processing, Cairo, Egypt, Nov. 2009, pp. 125 –128.

[12] BOEHM, W.: 'On cubics: a survey', ibid., 1982, 19, pp. 201-226

[13] SZELISKI, R.: 'Real time coding of hand drawn curves', M.Sc. Thesis, University of British Columbia, Vancouver, Canada 1981
[14] LAPALME, R.S.: "An interactive data reduction technique for line drawings', M.Eng. Thesis, Royal Military College, Kingston, Canada 1977
[15] DUBE, R.P.: 'Preliminary specification of spline curves', / EEE Trans; 1979,C-28, pp.286-290
[16] LIOU, M-L.: "Spline fit made easy', / EEE Trans., 1976, C-25, pp. 522- 527
[17] MOSS, R., and LINGARD, A.: "Parametric spline curves in integer Arithmetic designed for use in microcomputer controlled plotters', Computer. & Graphics, 1979, 4, pp. 51—6
[18] A generalized Fibonacci LSB Data Hiding Technique F.Battisti,Tampere University of Technology, Tampere,Finland.

AUTHORS

Mr. Allam Mohan received the B.Tech. degree in Computer Science and Engineering from JNTUH University, Hyderabad, India .He is having 4 years of teaching experience. Currently, he is working with Shri Vishnu Engineering College for Women Bhimavaram, as an Assistant Professor. His research interests are Digital Image Processing, Neural Networks and Biometrics.

Mr. Kampalli Ramu received the B.Tech. degree in Computer Science and Engineering from JNTUH University, Hyderabad, India and M.Tech degree in Computer Science Engineering from Osmania University, Hyderabad. He is having 4 years of teaching experience. Currently, he is working with Adam's Engineering College**,** paloncha, as an Assistant Professor. His research interests are Digital Image Processing and Operating Systems.

Mr. Salina Adinarayana received his M.E degree in Computer Science Engineering from Sathyabama University, Chennai and M.C.A from Andhra University, Visakhapatnam. He is having 14 years of teaching and 3 years of Research experience. He is a Certified Examiner for Quality management for IMC RBNQA 2011.He is a certified Mission 10X Learning Approach Practitioner. Currently, he is working with Shri Vishnu Engineering College for Women Bhimavaram. as an Associate Professor. His research interests are Knowledge Engineering, Object Oriented Technologies, Multimedia Mining and Data Structures.