Lossless Compression Scheme for Regular Images using Number Theoretic Transform

Salila Hegde
NIEIT, Mysuru
No 50(Part A), Hootagalli
Industrial area,
Koorgalli, Mysuru

Rohini Nagapadma
National Institute Of Engg.
Manandavadi Road,
Mysuru, India

ABSTRACT

In this paper properties of number theoretic transforms are investigated and it is found that they can be used to compress the regular image effectively. NTT variants namely Fermat and Mersenne transforms are applied on test images of size 16x16 and the results are analyzed and a compression scheme is developed. The algorithm is implemented in MATLAB and results are analyzed and compared with DCT in terms of total number of zero coefficients and total number of pixels in error when inverse transform is applied. The study shows that the transform is error free and can compress regular data effectively. Further investigations on these transforms are to be carried out and algorithms need to be developed to compress other images as well in order to achieve lossless compression.

General Terms

Image compression, Algorithms. Regular images

Keywords

Compression, NTT, Fermat, Mersenne, DCT, Rings, Fields.

1. INTRODUCTION

There is ever increasing demand for image compression due to the requirement of fast transmission and real time processing of images and videos over internet. Medical and satellite images require large storage size due to their high volumes and an efficient image compression scheme is desirable. There are several approaches on lossy and lossless compression of images during previous years. Lossless compression is preferred for medical images in clinical research and diagnosis. DCT and DWT based compression schemes are in wide use but they suffer due to the rounding error and also lossless schemes yield less compression ratio.

Number theoretic transforms are in use since seventies and found applications in several signal processing applications [1-5]. There are number of publications on properties of these transforms and their applications such as fast convolution and correlation, digital filtering , matrix multiplication, coding and decoding etc.[6-10] However the transform domain coefficients do not have specific meaning as in DCT. Due to this the use of NTT is mainly restricted to convolution, filtering and fast multiplication and less amount of work has been done in the field of image compression.

In this paper the research work is carried out on properties of 2D NTT and its variants namely Fermat and Mersenne number transforms. NTT is applied on regular images of size 16x16 and found that transform domain coefficients are also regular and many of the coefficients are zero. Making use of this property an image compression scheme is suggested.

Section I gives definition and properties of NTTS and its variants, Section II shows the result of 2D Fermat number transform 2DFNT of different images of size 16x16 and analyzes their results Section III explains an image compression scheme based on results of section II .Section IV shows the implementation and results. Section V discusses conclusion and future scope.

2. DEFINITION AND PROPERTIES OF NTT AND ITS VARIANTS

2.1 NTT Definition

NTTs are discrete Fourier transforms, defined over finite fields or rings. The NTT of a signal x (n) **and** its inverse INTT are defined as follows:

$$X(k) = \sum_{n=0}^{N-1} x(n)\,\alpha^{n/k} \bmod M, k = 0,1,2....N-1$$

Where M is some integer taken as a modulus. α is a root of unity of order N (i.e. $a^N = 1 \bmod F$). N is transform length. It is desirable but not necessary that the modulus M is a prime number. Readers are further referred to read about number theory and properties to understand the transform definition clearly. [11-12].

The two-dimensional NTT (2D-NTT) of an image x (m, n) **and** its inverse are given by

$$X(k,l) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} X(m,n)\,\alpha_1^{mk}\,\alpha_2^{nl} \bmod F$$

K=0, 1, 2, M-1 and l=0, 1, 2, N-1, Where F is modulus

The 1D and 2D inverse NTTs are given by

$$x(n) = (N^{-1}\sum_{k=0}^{N-1} X(k)\,\alpha^{nk}) \bmod M, n = 0,1,2.......N-1$$

And

$$x(m,n) = (N^{-1}\sum_{k=0}^{N-1} X(k,l)\,\alpha 1^{mk}\,\alpha 2^{nk}) \bmod F$$

$$n = 0,1,2......N-1, m = 0,1,2---M-1$$

Fermat and Mersenne number transforms are the twopromising NTTS among the variants.

2.2 Fermat number transforms FNT

This is one of the most promising NTT, where M is chosen as Fermat number Ft which can be prime or nonprime [13-15].

$M=F_t=2^b+1$, where $b=2^t$. , t=0, 1, 2….

F_t is termed as t^{th} Fermat number, F_0 through F_4 are prime and rest all are composite numbers.

The first five Fermat numbers which are prime are

t=0, $2^0=1$, $F_0=3$

t=1, $2^1=2$, $F_1=5$

t=2, $2^2=4$, $F_2=17$

t=3, $2^3=8$, $F_3=257$

t=4 , $2^4=16$, $F_4=65537$ and so on...

NTTs, which set modulus M as Fermat number, are called as Fermat number transform.

2.3 Mersenne number transform MNT

Rader proposed a number theoretic transform known as Mersenne number transform [16], where the modulus is chosen as

$M= M_p=2^p-1$ where p is prime.

The advantage of using these transforms is that value of α can be set to 2 and this is an attractive property when the transform is implemented using hardware all the multiplications can be performed using simple shift operations

The disadvantage is that there is restrictive constraint on transform length N and machine word size w. By using prime modulii, maximum transform length of N=w-1 is possible. However by using $\alpha = \sqrt{2}$ and using any α other than 2 longer transform lengths can be achieved.

2.4 Dimensional Ntt Of Images

2.4.1. Test images

Images of fabric texture, images of photo masks used in integrated chip manufacturing etc. are regular in their structure i.e., the intensity of picture elements is periodic. Figure 1 and 2 are examples for such images.

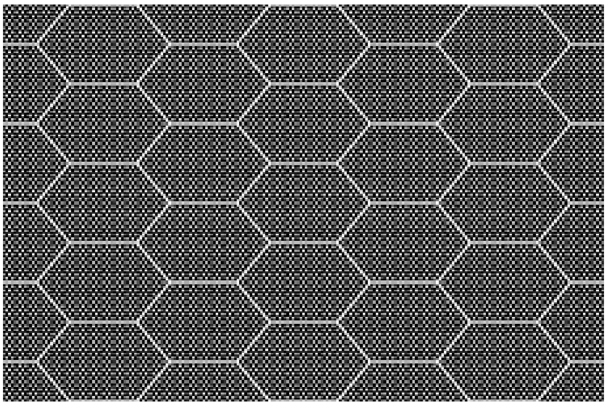

Fig 1: Image of fabric texture

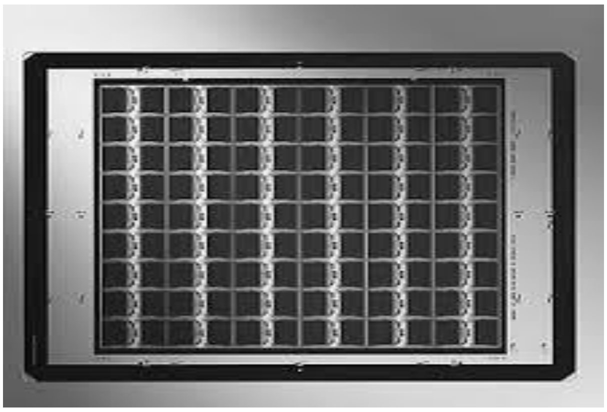

Fig 2: Image of photo mask

So to study the NTT of such regular images, test images of size 16X16 with repetitive pixels along rows and columns with periods Pr and Pc respectively are chosen. Figures 3a, 4a, 5a, 6a, 7a are images of period 8x8, 4x4, 2x2, 2x4,1x1 respectively.

18	17	16	15	14	13	12	11	18	17	16	15	14	13	12	11
17	16	15	14	13	12	11	10	17	16	15	14	13	12	11	10
16	15	14	13	12	11	10	19	16	15	14	13	12	11	10	19
15	14	13	12	11	10	19	01	15	14	13	12	11	10	19	01
14	13	12	11	10	19	01	05	14	13	12	11	10	19	01	05
13	12	11	10	19	01	05	06	13	12	11	10	19	01	05	06
12	11	10	19	01	05	06	07	12	11	10	19	01	05	06	07
11	10	19	01	05	06	07	08	11	10	19	01	05	06	07	08
18	17	16	15	14	13	12	11	18	17	16	15	14	13	12	11
17	16	15	14	13	12	11	10	17	16	15	14	13	12	11	10
16	15	14	13	12	11	10	19	16	15	14	13	12	11	10	19
15	14	13	12	11	10	19	01	15	14	13	12	11	10	19	01
14	13	12	11	10	19	01	05	14	13	12	11	10	19	01	05
13	12	11	10	19	01	05	06	13	12	11	10	19	01	05	06
12	11	10	19	01	05	06	07	12	11	10	19	01	05	06	07
11	10	19	01	05	06	07	08	11	10	19	01	05	06	07	08

Fig 3aTest image with period PrxPc=8x8

25	79	27	28	25	79	27	28	25	79	27	28	25	79	27	28
93	94	95	96	93	94	95	96	93	94	95	96	93	94	95	96
67	68	69	70	67	68	69	70	67	68	69	70	67	68	69	70
12	13	14	15	12	13	14	15	12	13	14	15	12	13	14	15
25	79	27	28	25	79	27	28	25	79	27	28	25	79	27	28
93	94	95	96	93	94	95	96	93	94	95	96	93	94	95	96
67	68	69	70	67	68	69	70	67	68	69	70	67	68	69	70
12	13	14	15	12	13	14	15	12	13	14	15	12	13	14	15
25	79	27	28	25	79	27	28	25	79	27	28	25	79	27	28
93	94	95	96	93	94	95	96	93	94	95	96	93	94	95	96
67	68	69	70	67	68	69	70	67	68	69	70	67	68	69	70

12	13	14	15	12	13	14	15	12	13	14	15	12	13	14	15
25	79	27	28	25	79	27	28	25	79	27	28	25	79	27	28
93	94	95	96	93	94	95	96	93	94	95	96	93	94	95	96
67	68	69	70	67	68	69	70	67	68	69	70	67	68	69	70
12	13	14	15	12	13	14	15	12	13	14	15	12	13	14	15

Fig4a:Test image with period PrxPc= 4x4

1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
7	8	7	8	7	8	7	8	7	8	7	8	7	8	7	8

Fig 5a; Test image with period 2x2

20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65
20	30	40	50	20	30	40	50	20	30	40	50	20	30	40	50
35	55	75	65	35	55	75	65	35	55	75	65	35	55	75	65

Fig 6a:Test image with period 2x4

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Fig 7a:Test image with period PrxPc=1x1

2.5. Selection of NTT parameters

Studies are carried out on selection of suitable modulus and transform length. The following tables list various Fermat and Mersenne numbers that are used in the work and maximum transform length possible.

Table 1.Fermat number transform parameters

T	Fermat number Ft	Transform Length used α =2 α=√2	Maximum Transform length N
2	17	8 16	16
3	257	16 32	256
4	65537	16 32	65536

Table2: Mersenne Number transform parameters

P	Mersenne number Mp	Transform length used	Root of unity α	Maximum transform length
7	127	14	2	126
11	2047	11	2	2046
13	8191	13	2	8190

However transform lengths in image and video processing based on block based algorithms are relatively short. In an image the pixel values range between 0 to 255, we found Fermat number transform with Ft=257 are suitable. And maximum transform length of 256 is possible, which is largely sufficient. Hence 257 is most suitable choice form modulus which is a Fermat prime. The chosen values are, α_1=2, α_2=2, and M=N=16 and Ft=F_3=257.As listed in table Mersenne number also may be employed.

2.6 Transformed images

2DFNT is performed on the test images using the above said parameters. Figures 3b,4b,5b,6b,7b are 2D Fermat number transforms (2DFNT) of the test images 3a,4a,5a,6a,7a respectively.

89	0	4	0	10	0	101	0	140	0	96	0	150	0	38	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	207	0	140	0	93	0	7	0	190	0	51	0	165	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	140	0	106	0	17	0	116	0	160	0	245	0	97	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101	0	93	0	17	0	256	0	205	0	180	0	211	0	67	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
140	0	7	0	116	0	205	0	166	0	238	0	237	0	47	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
96	0	190	0	160	0	180	0	238	0	198	0	44	0	76	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	0	51	0	245	0	211	0	237	0	44	0	142	0	84	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38	0	165	0	97	0	67	0	47	0	76	0	84	0	110	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig 3b:2DFNT of fig3a

219	0	0	0	84	0	0	0	52	0	0	0	17	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
149	0	0	0	204	0	0	0	180	0	0	0	53	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	204	0	0	0	180	0	0	0	53	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	204	0	0	0	180	0	0	0	53	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig4b:2DFNT of fig4a

124	0	0	0	0	0	0	0	129	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig5b:2DFNT of fig5a

18	0	0	0	196	0	0	0	68	0	0	0	76	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
204	0	0	0	146	0	0	0	194	0	0	0	106	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig6b:2DFNT of fig6a

256	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig 7b:2D FNT of fig 7a

It is observed that the FNT of the test images are also regular in structure and many of the coefficients become true zeros without round off errors, which turns out to be an attractive property for image compression.

In paper[17] authors have analyzed that there exist a relation between number of nonzero coefficients and period Pr and Pc and they formulate an equation to find nonzero coefficients and hence computations can be reduced in the forward and inverse transform algorithm. These properties are used to implement an image compression scheme which is discussed in next section.

3. IMAGE COMPRESSION SCHEME USING 2DFNT

2DFNT and 2DIFNT are performed on the test images and it is found that transform is reversible and error free. Period along row and columns Pr and Pc respectively are found from the transformed image and compressed image of size PrxPc is constructed. Fig 3c, 4c, 5c, 6c, 7c are compressed images respectively

89	4	10	101	140	96	150	38
4	207	140	93	7	190	51	165
10	140	106	17	116	160	245	97
101	93	17	256	205	180	211	67
140	7	116	205	166	238	237	4
96	190	160	180	238	198	44	76
150	51	245	211	237	44	142	84
38	165	97	67	47	76	84	110

Fig3c: Compressed image for fig3b

219	84	52	174
149	204	180	53
16	204	180	53
26	204	180	53

Fig4c: Compressed image for fig4b

124	129
3	0

Fig5c: Compressed image for fig5b

18	196	68	76
204	146	194	106

Fig6c: Compressed image for fig6b

256

Fig7c: Compressed image for fig7b

Since 257 is chosen as modulus, the range of coefficients in the compressed image is 0 to 256. If we use 8bit representation the coefficient 256 would be represented as 0. So when IFNT is applied on the decoded image the result is obviously unexpected and not reversible. To overcome this a bit either 0 or 1 is transmitted for each coefficient i.e. bit 0 for coefficients in the range 0 to 255 and for the coefficient bit 1 and its magnitude as 00.So two images of size PrxPc , one is binary image of zeros and ones and another of the transformed compressed coefficients. Hence for fig.1c FNT yields 64 nonzero coefficients and the proposed scheme extracts these nonzero coefficients from the transformed image. Hence compressed data sizes to 64 bytes plus binary image of zeros and ones of size 8x8 bit as in fig.8.

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Fig 8:Binary image of size 8x8 bits sent with fig 1c

And the algorithm is described as

Encoding:

1) Obtain 2DFNT of given test image of size 16x16.

2) Find Pr and Pc from the transformed image

3) Construct the compressed image from the transformed image replacing 256 as 0

4) Construct the binary image for each coefficient in the compressed image.

Decoding:

1) Determine Pr and Pc from the size of compressed image.

2) Decode the compressed image using Pr and Pc and binary image.

3) Apply 2DIFNT on the decoded image.

4. IMPLEMENTATION AND RESULTS

In paper [15] author suggests use of FNT for image compression and the scheme is to send the value of nonzero coefficient along with their position in the image. So that method consumes one extra byte for each nonzero byte. Proposed algorithm save this extra byte of memory by sending only nonzero coefficients along with binary bit image.

The proposed algorithm is implemented in MATLAB and tested on images given in figures 3a, 4a, 5a, 6a, 7a. And the result is compared with the suggested algorithm in paper [17] in terms of compression ratio. Table3 gives the comparison in terms of total bytes to be sent and compression ratio in percentage.

FNT is also compared with DCT which is used in JPEG and MPEG, universally accepted image and video compression standards. 2 dimensional FNT and DCT is applied on the given test images. And number of zero coefficients is found in both cases. After applying DCT, coefficients are rounded off. Conversely after IDCT also coefficients are rounded off. Because of this when the result of inverse DCT is compared with original image few coefficients are found in error. In FNT we obtain true zeros and there is no rounding performed. Table 4 gives the comparison.

Table3: Comparison of algorithms.

Test image	Total bytes to be stored after compression		Compression ratio in percentage	
	Algorithm in paper[15]	Proposed algorithm	Algorithm in paper[17]	Proposed algorithm

Fig 3a	128 Bytes	72Bytes	50	71.87
Fig4a	32 Bytes	18 Bytes	87.5	92.97
Fig5a	8 Bytes	4Bytes+4bits	96.87	98.24
Fig6a	16 Bytes	9 Bytes	93.75	96.87
Fig7a	2 Bytes	1Bytes+1bit	99.21	99.6

Table 4: Comparison between FNT and DCT

Image	Total No. of zero coefficients after in the forward transform		Total No of coefficients in error after inverse transform	
	DCT	FNT	IDCT	IFNT
Fig 3a	133	192	4	0
Fig4a	159	240	2	0
Fig5a	249	252	0	0
Fig6a	175	198	1	0
Fig7a	255	255	0	0

5. CONCLUSION AND FUTURE SCOPE

The research work is carried out using Mersenne number transform using the parameters given in Table 2. And the similar performance is achieved. From table 3 and 4 it is obvious that NTT based compression outperforms DCT and it gives higher compression ratio and the transform is lossless since there is no rounding off error as in DCT.For test image where all pixels are equal the 2DFNT gives only one nonzero coefficient whose value can be evaluated using NTT parameters. From this study it is believed that it is worth carrying out further study on NTT based compression schemes which may be applicable on medical images which are high volume images, where lossless compression is preferred. However NTT based algorithms require large number of computations in the software, hardware implementation of processor based on FNT yields optimized results. Since NTT based algorithms are extremely regular it is well suited for ASIC implementation and that will give beneficial results in terms of speed.

6. ACKNOWLEDGMENTS

This research is carried out at research Lab, NIE, Mysuru, INDIA. We would like to thank Dr.Narasimha Koulgoud, Prof, Department of ECE, NIE for his valuable suggestions.

7. REFERENCES

[1] B.Gold and C.M.Rader, 1969, Digital Processing of Signals.McGraw-Hill.

[2] R. Blahut, 1985,Fast algorithms for digital signal processing. Addison-Wesley Publishing Company.

[3] POLLARD, J.M,1971, 'The fast Fourier transform in a finite field', Math.Comput., 25, pp. 365-374.

[4] J. M. Pollard,1976, "Implementation of number-theoretic transforms,"Electron. Lett., pp. 378-379.

[5] AGARWAL, R.C, and COOLEY, J.W.,1977, 'New algorithms for digital convolution', *IEEE Trans.*, **ASSP-25,** pp. 106-124

[6] R. C. Agarwal and C. S. Burrus, 1975, "Number theoretic transforms to implement fast digital convolution," ***Proc.*** *ZEEE.* vol. **63,** PU.550.-560.

[7] E. Vegh and L. M. Leibowitz, 1975, "Fast complex convolution using number theoretic transforms," Naval Res. Lab., Washington, DC, NRL Rep. 7935.

[8] . S. Gudvangen, Hgskulen i Buskerud, AUI, EMR, Kongsberg, Norway , Practical applications of number theoretic transforms

[9] NUSSBAUMER, H.J.,1981,'New polynomial transform algorithms for multidimensional DFTs and convolutions', *IEEE Trans.,* ASSP-29, pp. 74-83

[10] REED, I.S., and TRUONG, T.K.,1979,'A new hybrid algorithm for computinga fast discrete Fourier transform', *IEEE Trans.,* **C-28,**pp. 487-492

[11] D.burton,1980,'Elementory number theory'.

[12] P. J. Nicholson, 1971, "Algebraic theory of finite Fourier transforms," J. Comput. Syst**.** Sci., vol. 5, pp. 524-547..

[13] AGARWALL, R.C, and BURRUS, C.S.,1974, 'Fast convolution usingFermat number transform with application to digital filtering', *ibid.,* ASSP-22, pp. 87-97

[14] Tuukka Toivonen,Janne Heikkila.,2006, Video filtering with fermat number theoretic transforms using residue number system. IEEE transactions on circuits and systems for video technology.vol16,No1,

[15] Richard Conway, 2006,Modified Overlap Technique Using Fermat and Mersenne Transforms.. IEEE Transactions on circuits and systems-II: Express Briefs, VOL. 53, NO. 8.

[16] RADER, CM.,1972, 'Discrete convolution via Mersenne transform', *IEEETrans.,* **C-21,** pp. 1269-1273

[17] S.Boussakta, A.Y.Md.Shakaff, F.Marir, A.G.J.Holt. 1988, Number Theoretic Transforms of Periodic Structures and their Applications.IEE Proceedings,Vol.135.Pt.G.No 2