Fractal Image Compression Based on Complex Exponent Moments and Fuzzy Clustering

Yong-yu WANG1,*, Zi-liang PING2, Zhi-lin ZHU1 and Yong-qiang WANG1

1School of Information and Electronic Engineering, Shandong Technology and Business University, China
2Century College of Post and Communication University of Beijing, China
*yongyuwang@sina.com

Keywords: Fractal image compression, Complex Exponent Moments(CEMs), Fuzzy clustering.

Abstract. The encoding in fractal image compressions are very time-consuming, because a large numbers of sequential search through a list of domains are needed to find the best match for a given range block. The Complex Exponent Moments (CEMs) are shift, rotation, scale and intensity distorted-invariant. This invariance can be used to match fractal image, and 2-D Fast Fourier Transform (FFT) algorithm is easily used to calculate CEMs. An effective fractal image compression based on CEMs and fuzzy clustering is proposed in this paper. Firstly, domain blocks are categorized using fuzzy c-mean-clustering approach. Then range blocks are compared to find the best domain blocks based on the CEMs. It shows in experimental results that the encoding is speed up with better performance in contrast with other fractal algorithms.

Introduction

Fractal image compression was proposed by Jacquin in 1990[1]. It is realized using local self-similarity in an image. By exploiting redundancy related to self-similarity in an image, a high compression ratio with high decoding quality can be achieved. Additionally, fractal coding has the advantages such as resolution independence and fast decoding in contrast with other image compression methods. So fractal image compression is a promising technique that has great potential to improve the efficiency of image storage and image transmission.

The fundamental hypothesis of the fractal image compression is based on the Partitioned Iteration Function System(PIFS), which employs the self-similarity of the image to achieve image compression. In order to encode an image, the most similar domain block has to be found in a large domain pool of each block. For the baseline method, the encoding process is time-consuming since a large amount of computations of similarity measurement are required to find the best match. More and more researches have been dedicated to speed up the block matching process. Many kinds of search methods were presented, such as classification methods, including the Fisher’s classification method [2], Hurtgen’s classification technique[3], Mario’method[4], fuzzy classification[5] and etc. and artificial Kohonen neural networks[6], clustering[7], local variances[8-9], features [10]and adaptive methods[11]. Most of these improvements are achieved to restrict the search space of the domain block pool in order to reduce the computation requirements of the best matching search.

The fractal image compression is achieved using self-similarity of the image. In order to encode an image, the most similar domain block in a large domain pool of each block has to be found. As the key part of the classification, the classification features should better describe the information of the whole and every part of the image, as well as the similarity degree of image blocks. The features are what, which directly determines the number of the classes and the accuracy of classification, then determines the fractal encoding time and the quality of the reconstructed image. Due to the orthogonal and complete properties of the kernel function, the Complex Exponent Moments (CEMs) are orthogonal and complete. The original image can be reconstructed by its CEMs. Besides the Complex Exponent Moments (CEMs) are shift, scale, rotation and intensity distorted-invariant and can be used as image features, which are best matching the self-similarity of an image.
In this paper, an effective fractal image compression is presented based on CEMs and Fuzzy clustering. The Complex Exponent Moments (CEMs) for each image blocks are firstly calculated as the image features, and the fuzzy clustering is performed to achieve a more accurate classification. In contrast with the baseline fractal coding algorithm, the proposed algorithm can speed up the encoding process greatly with better quality in reconstructed image.

**Complex Exponent Moments**

The moment theory has been developed for about half century. In 1962, M. K. Hu first presented the Geometric Moments (GMs) which are invariant under shift, rotation and scaling. However, GMs are not orthogonal and are not suitable for image reconstruction[12]. M.R. Teague first proposed the concept of invariant orthogonal moments in 1980[13], and proved that the orthogonal moments can be used to reconstruct the original image. Sheng et al proposed invariant orthogonal Fourier-Mellin Moments (OFMMs)[14], Z. L. Ping et al proposed a series of invariant orthogonal moments, such as Chebyshev-Fourier Moments (CHFMs)[15], Radial-Harmonic-Fourier Moments (RHFMs)[16] and Jacobi-Fourier Moments (JFMs)[17].

All those moments are orthogonal multi-distorted invariant and can be used to reconstruct the original image. RHFMs have the best performance among the above moments in terms of image reconstruction and noise sensibility[16]. In [17], Jacobi-Fourier Moments (JFMs) are proposed as a generic theoretical form of the above mentioned orthogonal multi-distorted invariant moments. Furthermore, the RHFMs can be transformed into CEMs and A FFT algorithm is proposed to calculate CEMs, which is faster and more accurate than the integral methods used to calculate RHFMs[18-19]. The Complex Exponent Moments (CEMs) are shift, scale, rotation and intensity distorted-invariant and can be used as a good features of a image to reconstruct image. Comparisons with RHFMs show that CEMs have higher quality and lower computational complexity on image reconstruction. The definition of CEMs is given in [18-19].

The definition of the Radial-Harmonic Fourier Moments (RHFMs) in a polar coordinate system is given as follow.

\[
\phi_{nm} = \frac{1}{2\pi} \int_{0}^{2\pi} \int_{0}^{1} f(r, \theta) T_n(r) \exp(-jm\theta) r dr d\theta
\]  

(1)

\[
T_n(r) = \begin{cases} 
\frac{1}{r} & \text{if } n = 0 \\
\frac{2}{r} \sin(n+1)\pi r & \text{if } n \text{ is odd} \\
\frac{2}{r} \cos n\pi r & \text{if } n \text{ is even} 
\end{cases}
\]  

(2)

According to Euler’s formula, the harmonic function set can be transformed to a complex exponential function set.

\[
A_k(r) = \frac{1}{\sqrt{r}} \exp(j2k\pi r)
\]  

(3)

The relationship between and is as follow,

\[
\begin{align*}
T_0(r) &= \frac{1}{\sqrt{2}} A_0(r) n = k = 0 \\
T_n(r) &= \frac{1}{2j} (A_k(r) - A_{-k}(r)) n = 2k - 1 = 1,2,3 \\
T_n(r) &= \frac{1}{2} (A_k(r) + A_{-k}(r)) n = 2k = 1,2,3
\end{align*}
\]  

(4)
The Radial-Harmonic-Fourier Moments (RHFMs) can be rewritten as follow.

\[ E_{mn} = \frac{1}{4\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \frac{2}{r_{xy}} \exp(-j2k\pi r_{xy}) \exp(-jm\theta_{xy}) dx dy \]  

(5)

Here, \( r_{xy} = \sqrt{x^2 + y^2} \), \( \theta_{xy} = \arctan \frac{y}{x} \).

\( E_{mn} \) is defined as the Complex Exponent Moments (CEMs). Because of the perpendicularity and completeness of the kernel functions, the moments are orthogonal and complete moments inside the unit circle. The original image can be reconstructed by CEMs. CEMs can be rewritten in a Cartesian coordinate system.

\[ E_{mn} = \frac{1}{M} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} F_u(r_u, \theta_u) \sqrt{r_u} \exp(-j \frac{2\pi}{M} ku) \exp(-j \frac{2\pi}{M} mv) \]  

(6)

The Complex Exponent Moments in equ. (6) can be calculated using a 2-D Fourier transform (FFT), which is of lower complexity and more accurate.

**Fuzzy Clustering**

Fuzzy clustering is one of the common methods used frequently in clustering applications. The most famous method of fuzzy clustering is the fuzzy c-means method (FCM)[20]. FCM introduces the concept of fuzzy sets to classic K-means[21]. The FCM algorithm assigns domain blocks to each category by using fuzzy memberships. Usually, membership functions are defined based on a distance function, such that membership degrees express proximities of entities to cluster prototypes. Let denotes the domains pool with \( N \) blocks where \( \mathbf{x} \) represents the features data. To partition the domains pool into \( K \) clusters, an iterative optimization algorithm was used that minimizes the cost function defined as follows

\[ J = \sum_{i=1}^{c} \sum_{k=1}^{N} \mu_{ik}^m \| x_k - p_i \|^2 \]  

(7)

Where, \( \mu_{ik} \) represents the membership of blocks \( x_k \) in the \( i \) th cluster, \( p_i \) is the \( i \) th cluster center. FCM algorithm is described as follows.

**Initialization**: choose the clustering number \( c \), \( 2 \leq c \leq n \), \( n \) is the number of data, and the termination parameter \( \epsilon \), initiate the cluster center \( p^{(0)} \), set the iterative index \( b = 0 \).

**Step 1**: The membership functions \( \mu_{i}^{(b)} \) are updated as follows:

For \( \forall i,k \), if \( \exists d_{ik}^{(b)} > 0 \), then

\[ \mu_{ik}^{(b)} = \left[ \sum_{j=1}^{c} \left( \frac{d_{ik}^{(b)}}{d_{jk}^{(b)}} \right)^{\frac{1}{\mu}} \right]^{-1} \]  

(8)

if \( \exists i,r \), here \( d_{ir}^{(b)} = 0 \), then

\[ \mu_{ir}^{(b)} = 1 \], and \( j \neq r, \mu_{ij}^{(b)} = 0 \)  

(9)

**Step 2**: the cluster center \( p^{(b+1)} \) are updated as follows:

\[ p_i^{(b+1)} = \frac{\sum_{k=1}^{N} \left( \mu_{ik}^{(b+1)} \right)^m \cdot x_k}{\sum_{k=1}^{N} \left( \mu_{ik}^{(b+1)} \right)^m}, \quad i = 1,2,\ldots,c \]  

(10)
Step 3: if \( \| p^{(b)} - p^{(b+1)} \| < \varepsilon \), stop the iteration and output the membership function \( U \) and cluster center \( P \), otherwise \( b = b + 1 \), goto step 1.

**Proposed Method Based on CEMs and Fuzzy Clustering**

Firstly, D blocks are classified using fuzzy clustering. Then we find the D blocks with smallest distance for each R blocks. The distance is calculated by Equ.(11).

\[
d(Q', Q) = \sum |Q'_i - Q'_j|
\]

The proposed method is summarized as follows.
1) To partition the given image into R blocks and D blocks;
2) To calculate 4×4 CEMs for every R blocks and D blocks;
3) To apply fuzzy clustering to D blocks;
4) To calculate the distance from center of each cluster in a D blocks for every R blocks, and select the smallest distance and record the best matched D blocks;
5) For each R blocks, the sequence number of the corresponding symmetrical transformations, the scaling coefficients and the luminance offset can be obtained to get fractal image encoding.

**Experimental Results and Discuss**

A 512×512 Lena image was used for testing and validating the efficiency of the above algorithm. The experiments were conducted on a notebook computer with a 2.0GHZ CPU, 256 MB memory. The operating system is Windows 8, and the simulation environment is MATLAB. In the experiments, the baseline fractal coding algorithm, the fractal coding based on CEMs and the proposed fractal coding algorithm were conducted in order to make comparing. Furthermore, the three kind of dividing modes were adopted respectively to evaluate the effect of dividing number. The time consuming in coding process were listed in Table. 1.

**Comparison of Image Quality**

The performance is demonstrated Fig.1, Fig. 2, and Fig.3, including the original image, decoding image under R blocks with 4×4 and D blocks with16×16, decoding under R blocks with4×4 and D blocks with 32×32, and decoding image under R blocks with16×16 and D blocks with 32×32 from left to right.

1) Image using baseline fractal coding algorithm

![Figure 1. original image and decoding image using baseline fractal algorithm.](image1)

2) Image using fractal coding based on CEMs
Comparison of Computation Time

The computation time of three algorithms are listed in table 1.

<table>
<thead>
<tr>
<th>block</th>
<th>baseline Coding time</th>
<th>Decoding time</th>
<th>With CEMs Coding time</th>
<th>Decoding time</th>
<th>Our method Coding time</th>
<th>Decoding time</th>
</tr>
</thead>
<tbody>
<tr>
<td>R D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4x4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8x8</td>
<td>1304.904799</td>
<td>10.006054</td>
<td>9.353784</td>
<td>12.172923</td>
<td>7.260280</td>
<td>12.213362</td>
</tr>
<tr>
<td>16x16</td>
<td>327.992074</td>
<td>2.856080</td>
<td>4.997056</td>
<td>3.501404</td>
<td>4.379705</td>
<td>3.520140</td>
</tr>
<tr>
<td>32x32</td>
<td>82.169381</td>
<td>1.084781</td>
<td>4.245138</td>
<td>1.567395</td>
<td>4.524350</td>
<td>1.553382</td>
</tr>
<tr>
<td>64x64</td>
<td>20.823062</td>
<td>0.607026</td>
<td>4.006862</td>
<td>0.990693</td>
<td>4.268062</td>
<td>1.080373</td>
</tr>
<tr>
<td>128x128</td>
<td>5.514446</td>
<td>0.523314</td>
<td>4.104307</td>
<td>0.979747</td>
<td>1.121020</td>
<td>0.995075</td>
</tr>
<tr>
<td>8x8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32x32</td>
<td>22.250545</td>
<td>2.506849</td>
<td>2.270299</td>
<td>2.872421</td>
<td>2.389293</td>
<td>2.869357</td>
</tr>
<tr>
<td>64x64</td>
<td>5.825374</td>
<td>0.726353</td>
<td>1.582101</td>
<td>1.026393</td>
<td>1.823561</td>
<td>1.152945</td>
</tr>
<tr>
<td>128x128</td>
<td>1.778527</td>
<td>0.302374</td>
<td>1.459520</td>
<td>0.596322</td>
<td>1.775202</td>
<td>0.589352</td>
</tr>
<tr>
<td>16x16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>64x64</td>
<td>2.565065</td>
<td>2.449505</td>
<td>1.581043</td>
<td>2.756048</td>
<td>1.778105</td>
<td>2.605085</td>
</tr>
<tr>
<td>128x128</td>
<td>0.924745</td>
<td>0.715888</td>
<td>0.947727</td>
<td>0.933632</td>
<td>1.275602</td>
<td>0.998572</td>
</tr>
</tbody>
</table>

In baseline fractal image algorithm, the performance of decoding image is good when R blocks with 4x4 and the best performance is D blocks with 16x16. This algorithm is obviously an extraordinarily time-consuming process. For algorithm with CEMs, it also shows that the coding and decoding time are reduced significantly with better quality of the decoding image. The proposed algorithm with the R blocks 4x4 and D blocks 8x8 or 16x16, and R blocks 8x8 and D blocks 16x16 can reduce coding and decoding time, and the quality of decoding image is improved compared with the algorithm based on CEMs.

Conclusion

An effective fractal image compression is presented based on Complex Exponent Moments (CEMs) and Fuzzy clustering in this paper. The CEMs of each image blocks are firstly calculated as the image features, and the fuzzy clustering method is used to achieve a more accurate classification. Experiment results show that the proposed method can speed up the encoding process greatly and achieve better performance in quality of decoding image. The proposed algorithm is a desirable
choice for fractal image encoding when considering the encoding time, the quality of the decoding image and the complexity of the algorithm at the same time. Furthermore, many existing complicated reduction techniques can be incorporated into this proposed algorithm to achieve better performance.

References


