

Contents

- [变量的初始化](#)
- [张量乘法定义计算](#)
- [快速傅立叶变换方法计算](#)
- [构造张量的循环矩阵](#)
- [构造张量的向量形式矩阵](#)
- [恢复张量形式](#)

```
clear
clc
```

变量的初始化

A 大小为 (n_1, n_2, n_3) , B 大小为 (n_2, l, n_3) , $R = A \times B$, R 的大小为 (n_2, l, n_3)

```
n1 = 48; % 48行
n2 = 580; % 580张
n3 = 42; % 42列
l = n2;
img_size = [n1, n2, n3];
A = rand(img_size);
B = rand(n2, l, n3);
times = 1; % 重复计算次数
```

张量乘法定义计算

```
tic
n = 0;
while n < times
    circA = circ(A);
    MatVecB = MatVec(B);
    temp = circA * MatVecB;
    r = fold(temp, n1, n3);
    n = n + 1;
end
disp('使用张量乘法的定义计算耗时: ');
toc
```

快速傅立叶变换方法计算

```
tic
n = 0;
while n < times
    FFT = zeros(n1, l, n3);
    X = fft(A, [], 3);
    Y = fft(B, [], 3);
    for i = 1:n3
        FFT(:, :, i) = X(:, :, i) * Y(:, :, i);
    end
    Lb = ifft(FFT, [], 3);
    n = n + 1;
end
```

```
end
disp('使用快速傅立叶变换方法计算耗时: ');
toc
```

使用快速傅立叶变换方法计算耗时:
时间已过 0.723135 秒。

构造张量的循环矩阵

```
function [circM]=circ(M)
    [x, y, z] = size(M);
    circM = zeros(x*z, y*z);
    index = gallery('circul',1:z)';
    for i=1:z
        for j=1:z
            slice = index(i,j);
            frontalSlice = M(:,:,slice);
            circM((i-1)*x+1:i*x, (j-1)*y+1:j*y) = frontalSlice;
        end
    end
end
```

构造张量的向量形式矩阵

```
function [MatVecM]=MatVec(M)
    [x, y, z] = size(M);
    MatVecM = zeros(x*z, y);
    for i=1:z
        MatVecM((i-1)*x+1:i*x, :) = M(:,:,i);
    end
end
```

恢复张量形式

```
function t = fold(M, x, z)
for i=1:z
    value = M((i-1)*x+1:i*x, :);
    t(:,:,i) = value; %#ok<AGROW>
end
end
```

使用张量乘法的定义计算耗时:
时间已过 4.255921 秒。