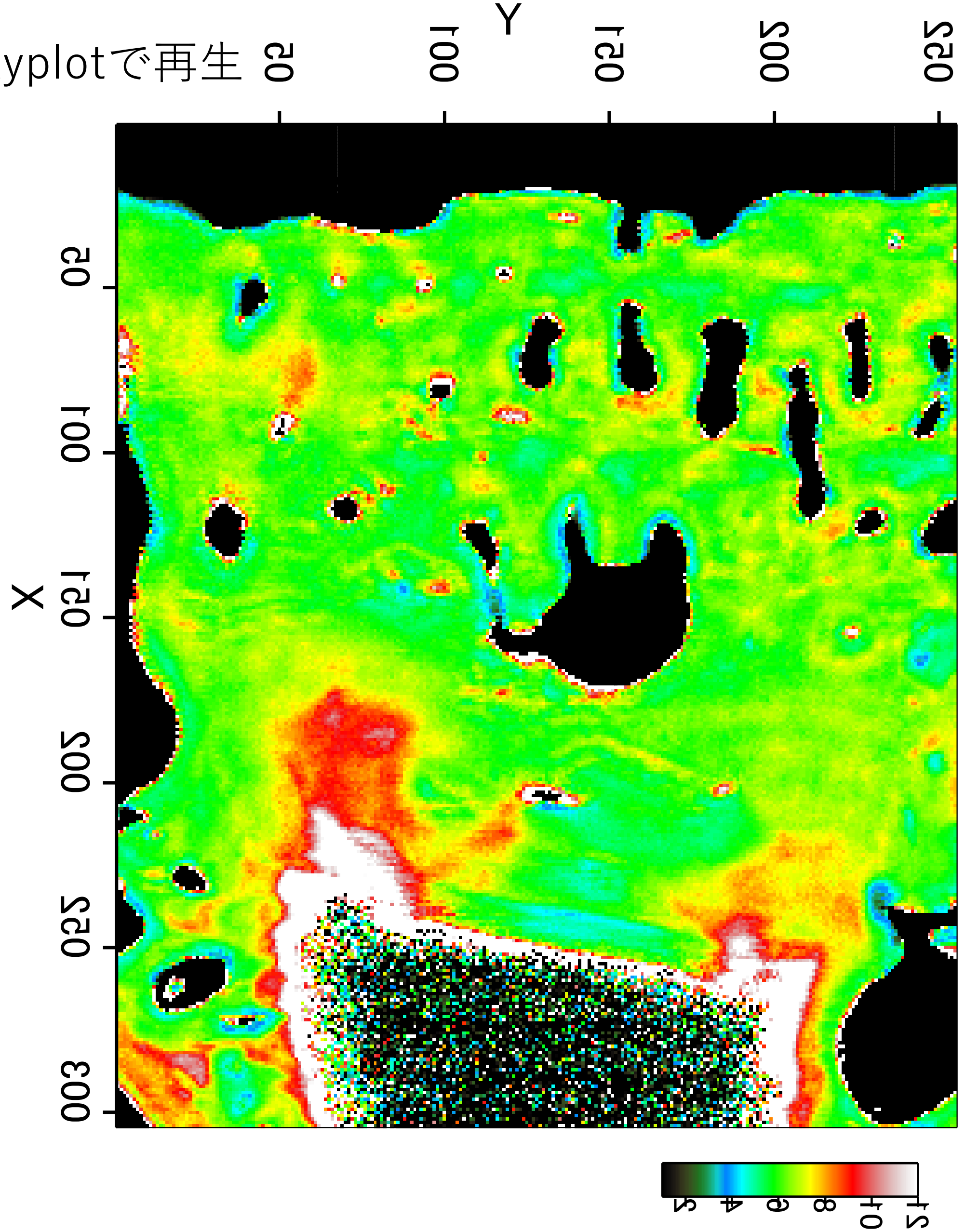
readBinary_double.c

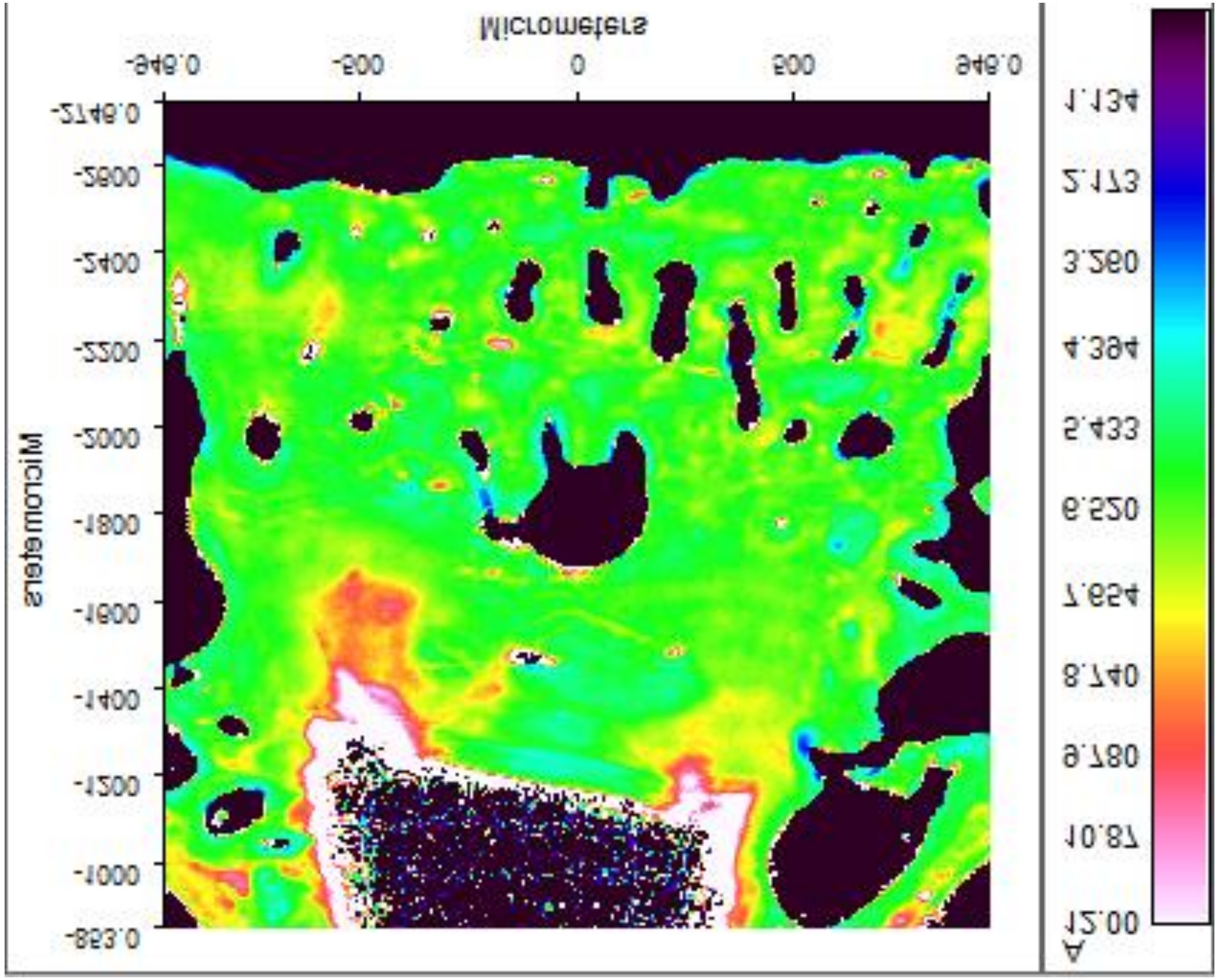
```c
#include <stdio.h>
#include <stdlib.h>
#define READ_SIZE 1
#define SKIP_SIZE 18

int main(int argc, char* argv[]){
    FILE * fp = NULL;
    int i = 0;
    int j = 0;
    double data;
    int delimiter = READ_SIZE*atoi(argv[3]);
    size_t ret;

    fp = fopen(argv[1], "rb");
    int i_max = atoi(argv[2]);

    for (i=0; i < SKIP_SIZE; i++) ret = fread(&data, sizeof(double), 1, fp);
    for (i=SKIP_SIZE; i < atoi(argv[2]); i++){
        ret = fread(&data, sizeof(double), 1, fp);
        if(data<0) data=0;
        printf("%f¥t",data);
        if((i-SKIP_SIZE)%delimiter==0) printf("¥n");
    }
    fclose(fp);
    return 0;
}
```

```
.¥readBinary_double.exe .¥sample.imp 92434 304 > double.txt
```

Kyplotで再生



オリジナル画像

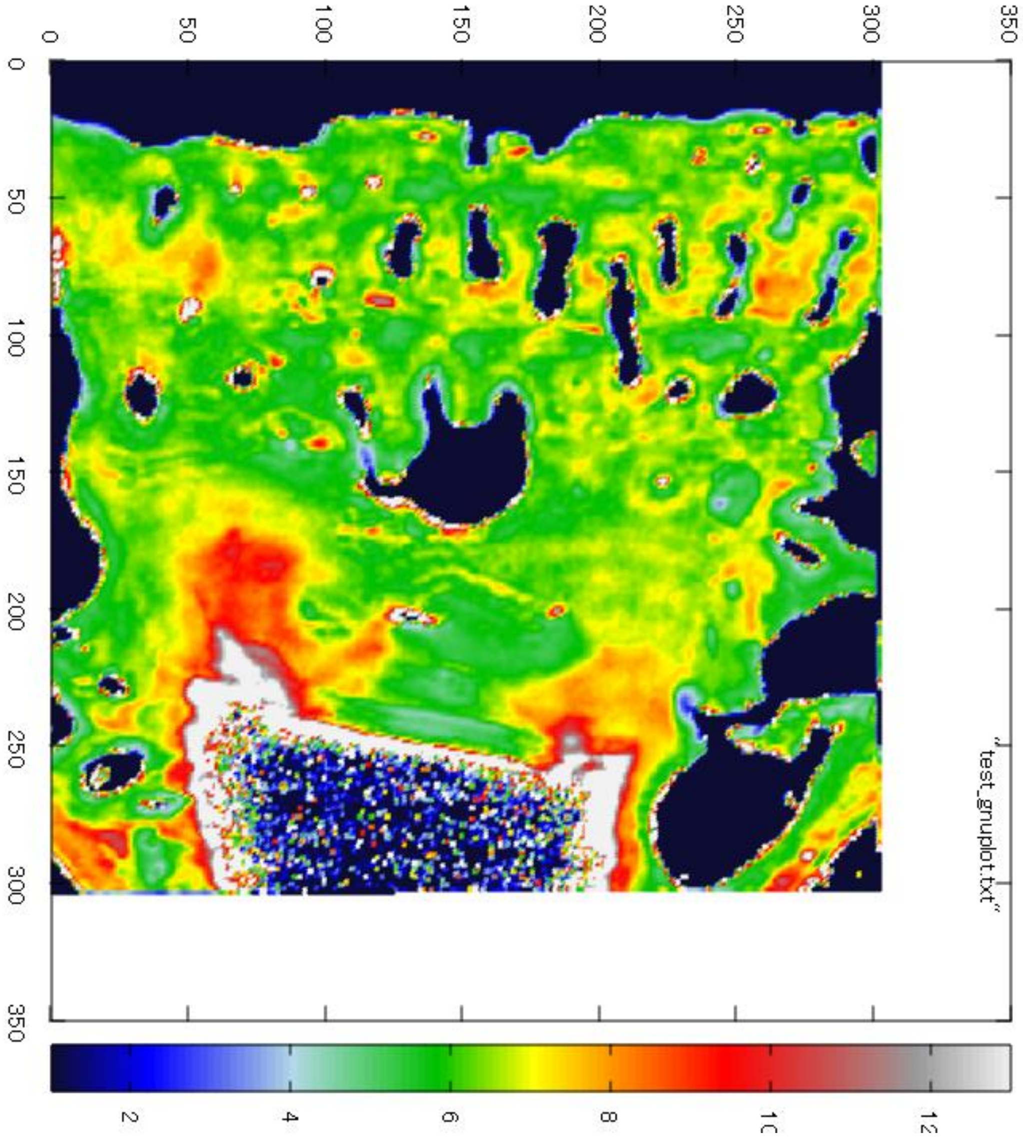


readBinary_double_gnuplot.c

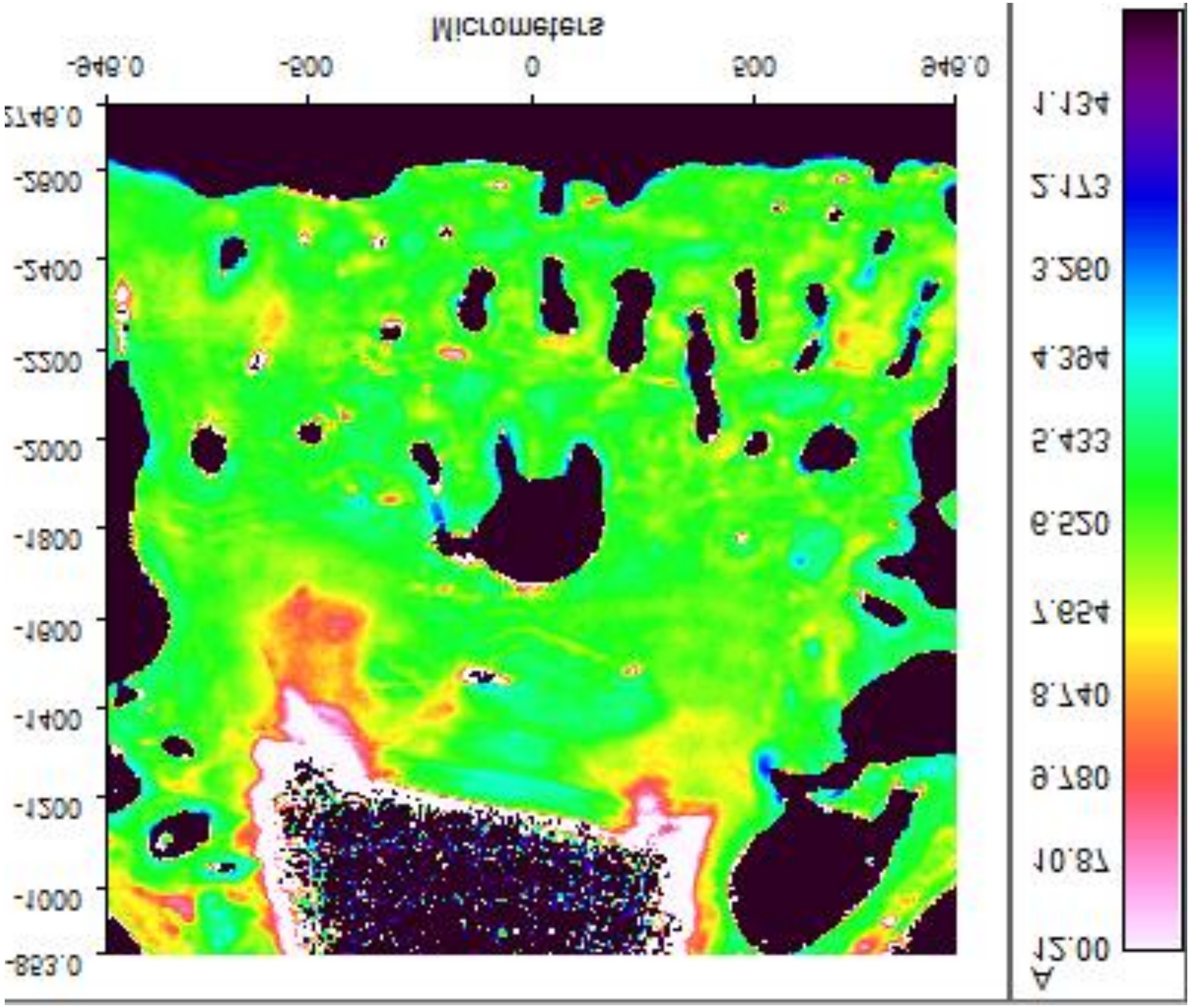
```c
#include <stdio.h>
#include <stdlib.h>
#define READ_SIZE 1
#define SKIP_SIZE 18

int main(int argc, char* argv[]){
    FILE * fp = NULL;
    long i = 0;
    long j = 0;
    double data;
    j = READ_SIZE*atol(argv[3]);
    size_t ret;

    fp = fopen(argv[1], "rb");
    long i_max = atol(argv[2]);

    for (i=0; i < SKIP_SIZE; i++) ret = fread(&data, sizeof(double), 1, fp);
    for (i=SKIP_SIZE; i < atol(argv[2]); i++){
        ret = fread(&data, sizeof(double), 1, fp);
        if(data<0) data=0;
        printf("%d %d %f¥n",(i-SKIP_SIZE)/j, (i-SKIP_SIZE)%j, data);
        if((i-SKIP_SIZE)%j==j-1) printf("¥n");
    }
    fclose(fp);
    return 0;
}
```

```
.¥readBinary_double_gnuplot.exe .¥mid.imp > mid_gnuplot.txt
```

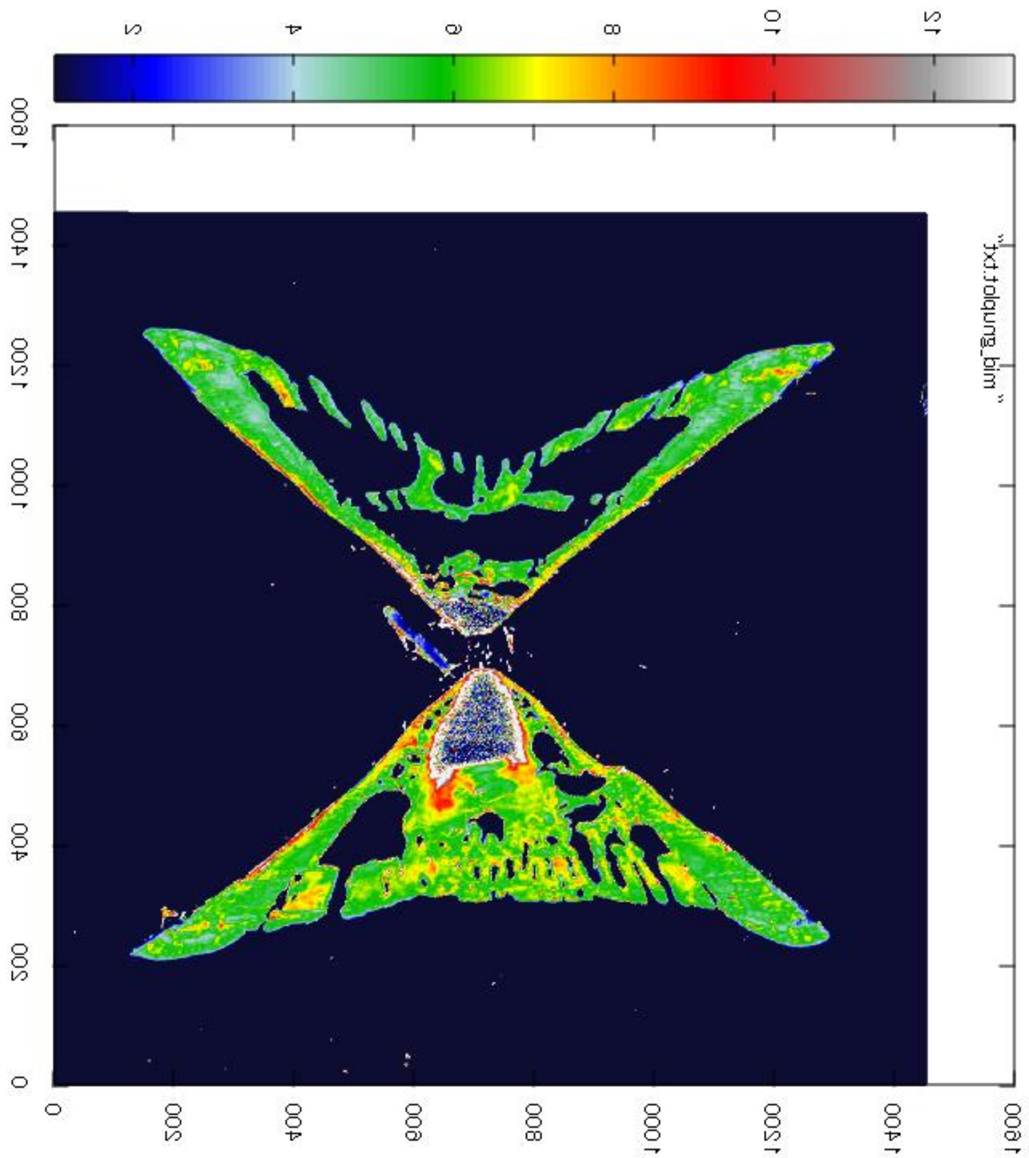
gnuplotで再生



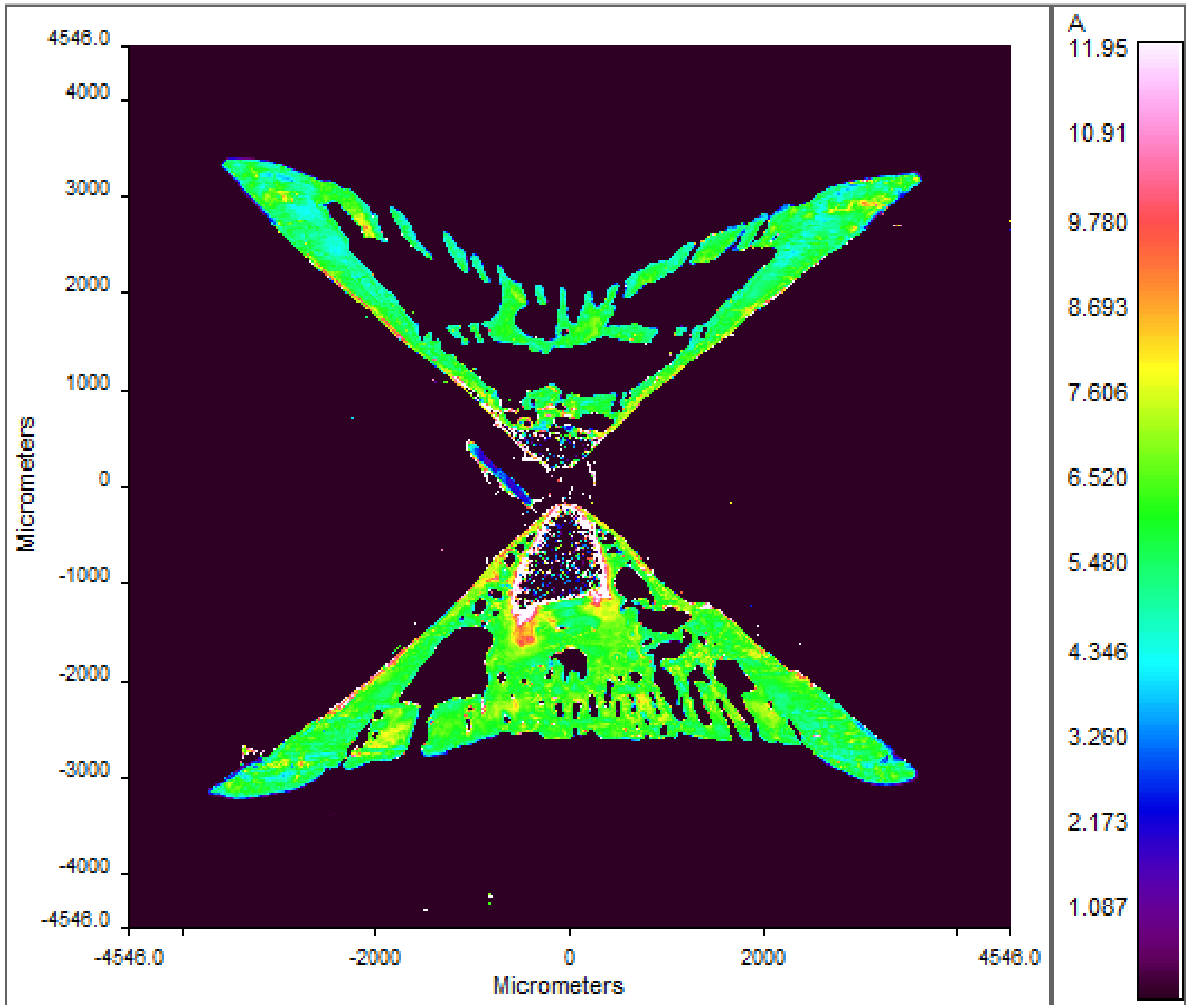
オリジナル画像



gnuplotで再生



オリジナル画像


Micrometers
Micrometers
A