

# ImageJ で空間周波数フィルタリング

以前使っていた、ScionImage では意外に簡単だった空間周波数フィルタリングだが、ImageJ ではどのように操作するのか本格的にやってみるまで試したことが無かった。今回ゼミ生の卒研のために、ImageJ でできれば楽だなと思いやってみることにした。

## 用意するもの

ImageJ

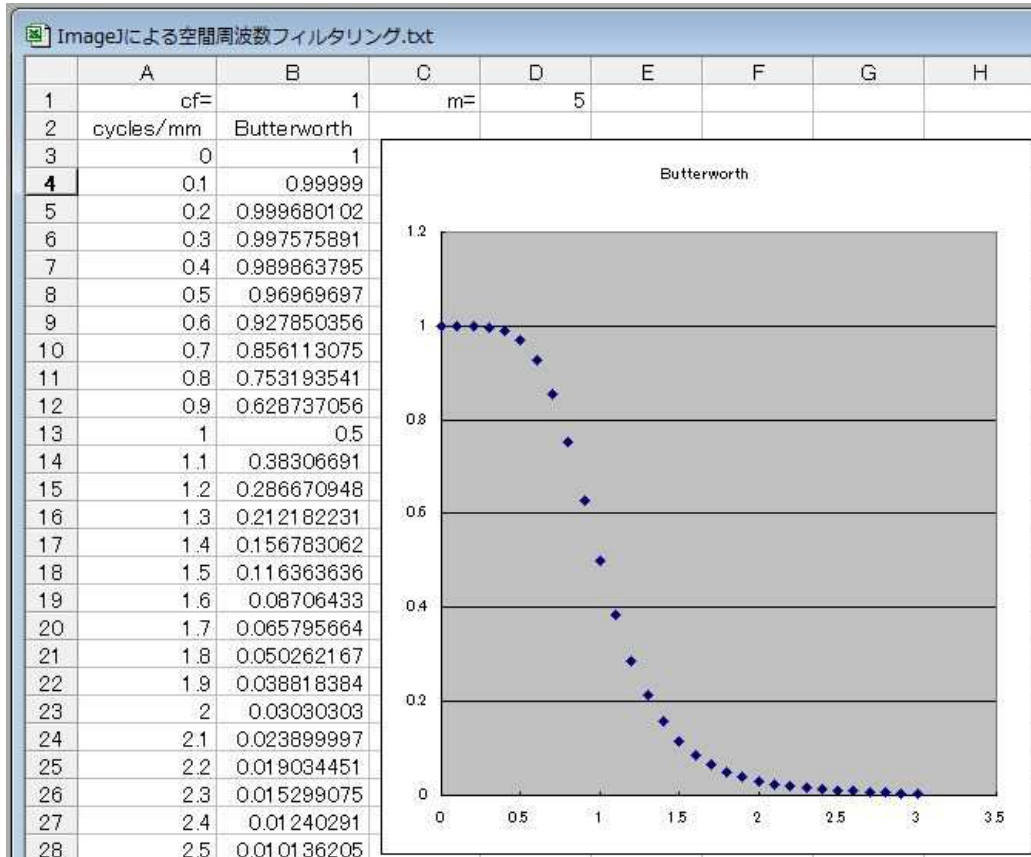
マイクロソフト Excel

フィルタリングのための関数（MTF のように 1 次元のもの）

※ 1 次元の関数を 2 次元に拡張してフィルタ関数として使用する。

## 手順 1 Excel 上での作業

- ① 空間周波数フィルタの 1 次元関数を求める。今回はカットオフ周波数 1 オーダー 5 のバターワースフィルタを使用した。



- ② 空間周波数フィルタを 2 次元に展開する。  
半径や関数および画像上のマトリックス数などを決定する。

x 軸の場合

The figure shows an Excel spreadsheet with columns A through L. It details the parameters for a 2D filter expansion. The radius is 3, and the function is  $1/(1+(x/1)^5)$ . The matrix size is 64x64. A table shows the values of the function for x from 31 down to 28, with the first row (x=31) containing values from 2.91 to 2.06. The second row (x=30) contains zeros.

A	B	C	D	E	F	G	H	I	J	K	L
半径 r=	3	関数 = $1/(1+(x/1)^5)$									※ 空
半径をセル数で割った値 =	0.09	マトリックスの大きさは 64 × 64									
		31	30	29	28	27	26	25	24	23	22
		=C3*\$F2	2.72	2.63	2.53	2.44	2.34	2.25	2.16	2.06	1
5	31	2.91	0	0	0	0	0	0	0	0	0
6	30	2.81	0	0	0	0	0	0	0	0	0
7	29	2.72	0	0	0	0	0	0	0	0	0
8	28	2.63	0	0	0	0	0	0	0	0	0

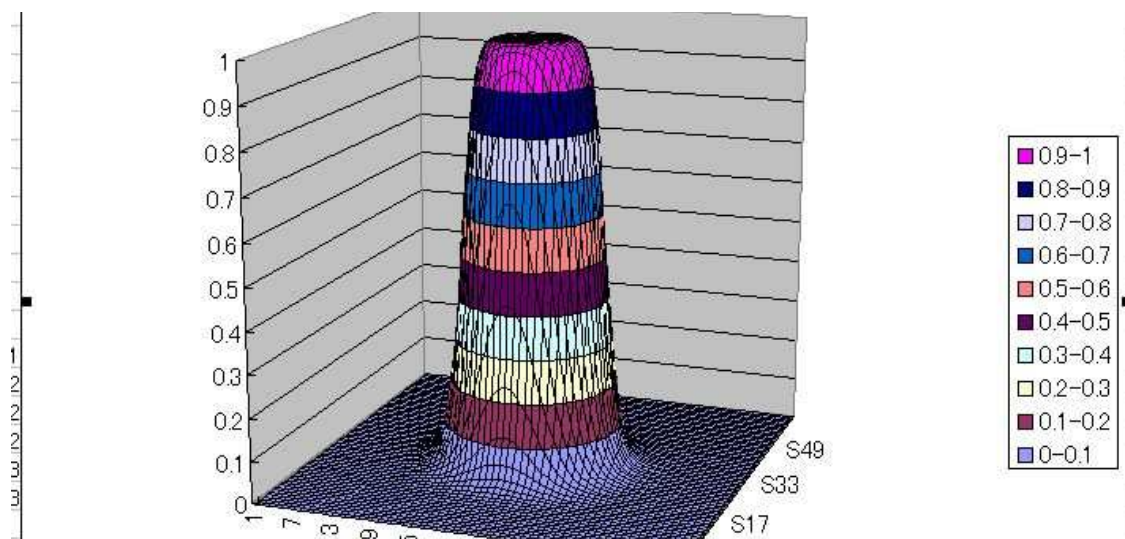
y 軸の場合

	A	B	C	D	E	F	G	H	I	J	K	L
1	半径 r= 3		関数= $1/(1+(x/1)^5)$									※ 空
2	半径をセル数で割った値=		0.09									マトリックスの大きさは64×64
3			31	30	29	28	27	26	25	24	23	22
4			2.91	2.81	2.72	2.63	2.53	2.44	2.34	2.25	2.16	2.06
5	31	=A5*B\$2	0	0	0	0	0	0	0	0	0	0
6	30	2.81	0	0	0	0	0	0	0	0	0	0

2次元に展開する。

	A	B	C	D	E	F	G	H	I	J	K	L
1	半径 r= 3		関数= $1/(1+(x/1)^5)$									※ 空
2	半径をセル数で割った値=		0.09									マトリックスの大きさは64×64
3			31	30	29	28	27	26	25	24	23	22
4			2.91	2.81	2.72	2.63	2.53	2.44	2.34	2.25	2.16	2.06
5	31	2.91	=IF(C\$4^2+\$B5^2<\$C\$1^2,1/(1+(C\$4^2+\$B5^2)^5),0)									
6	30	2.81	0	0	0	0	0	0	0	0	0	0

2次元に展開したバターワースフィルタ。

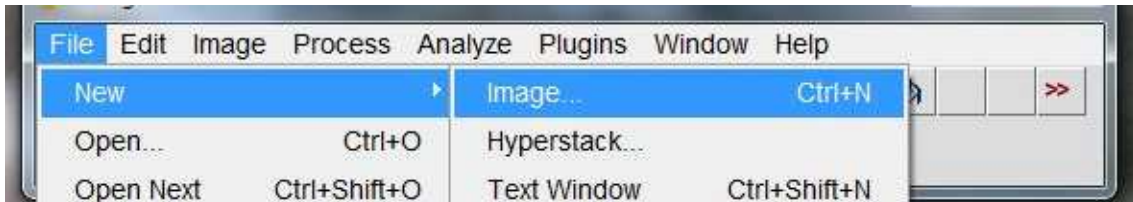


③ 計算結果をテキストファイルとして保存する。

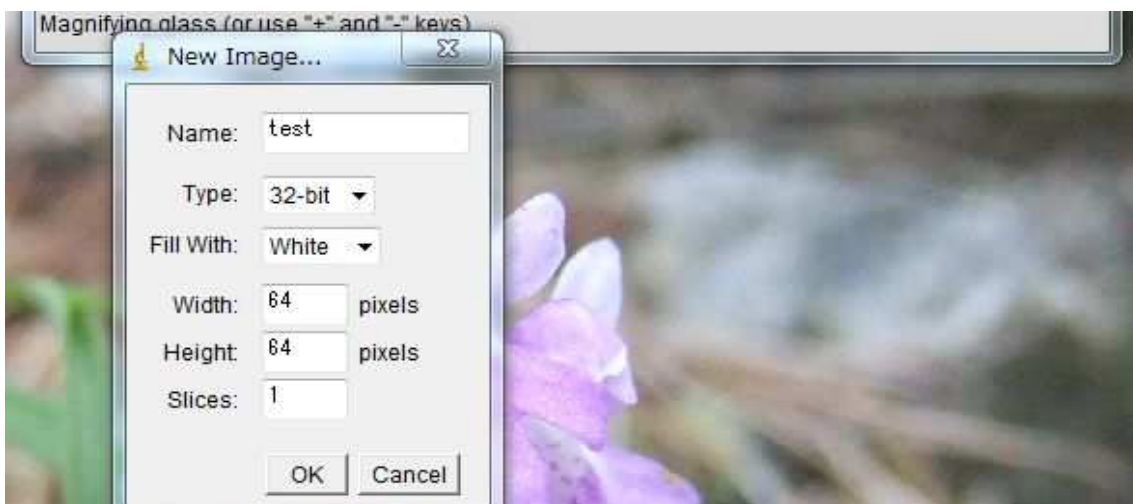
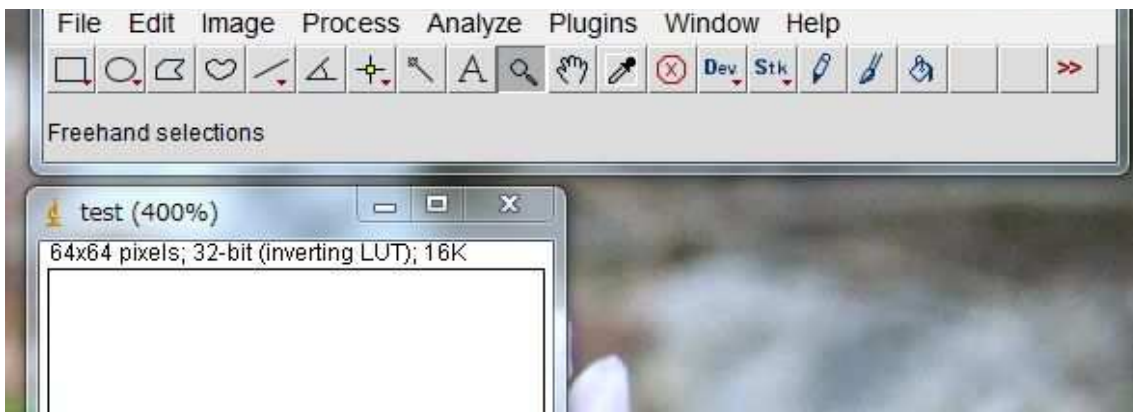


## 手順2 ImageJ での作業

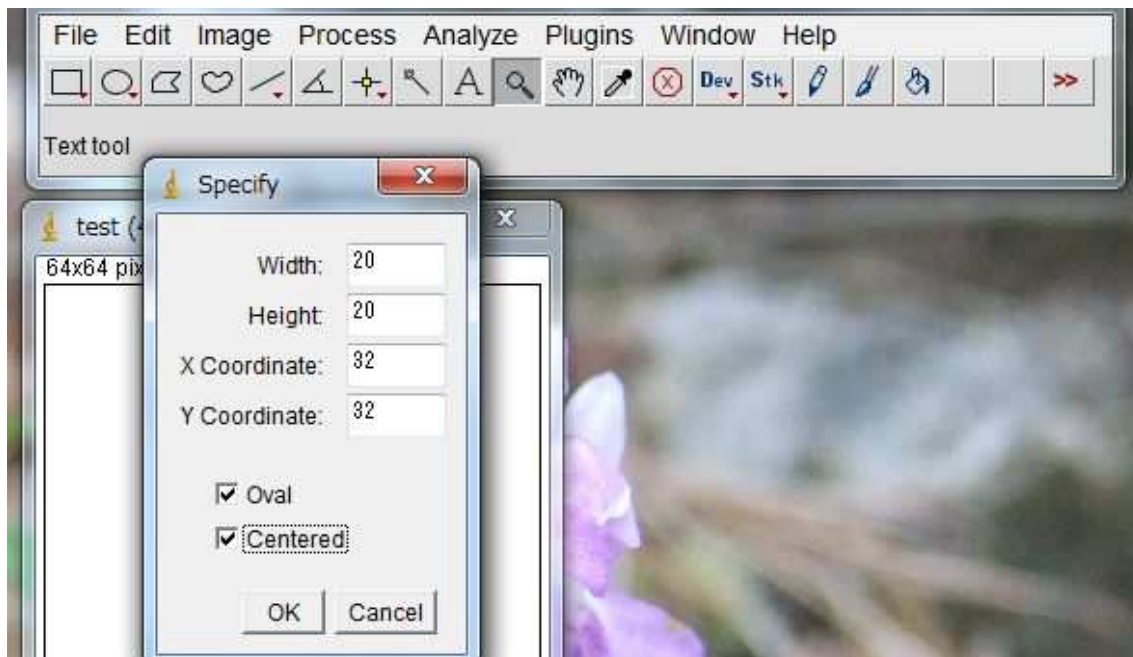
- ① 新しい image を作成する。(名前は適当に「test」とする)



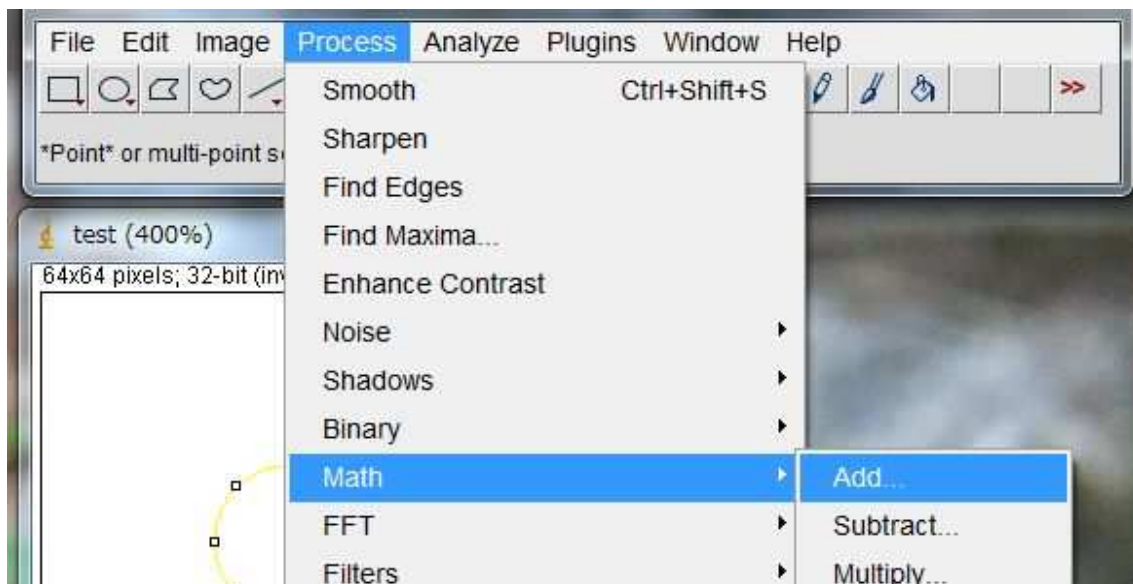
- ② 「New」 → 「Image」 は2のべき乗のマトリックスサイズで 32-bit で作成する。

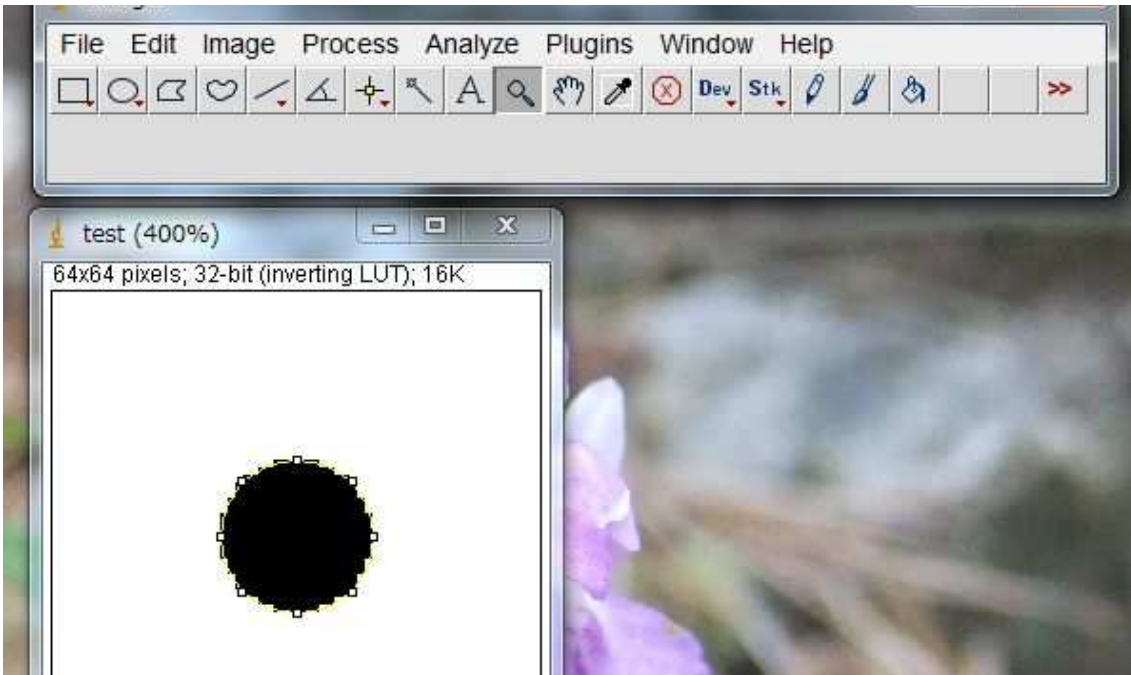
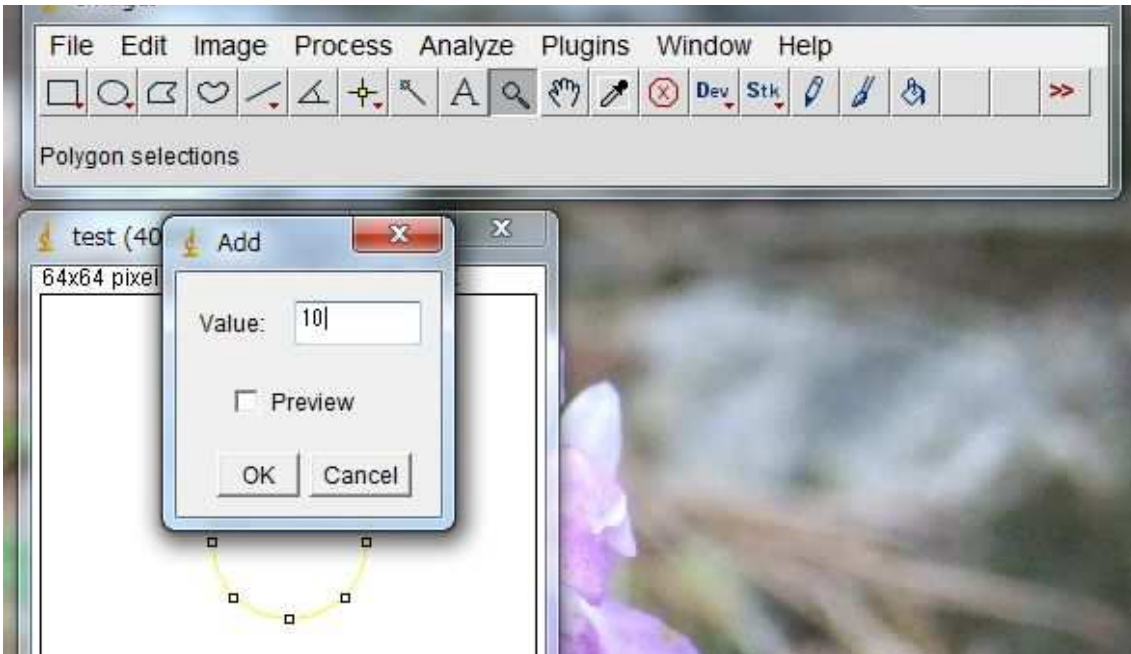


- ③ 今回は test Image として直径 20 の円形でピクセル値 10 のものを作成する。「Selection」→「Specify」メニューにて幅高さとも 20 を指定する。X 軸 Y 軸は画像の中心を指定して「Oval」もチェックする。

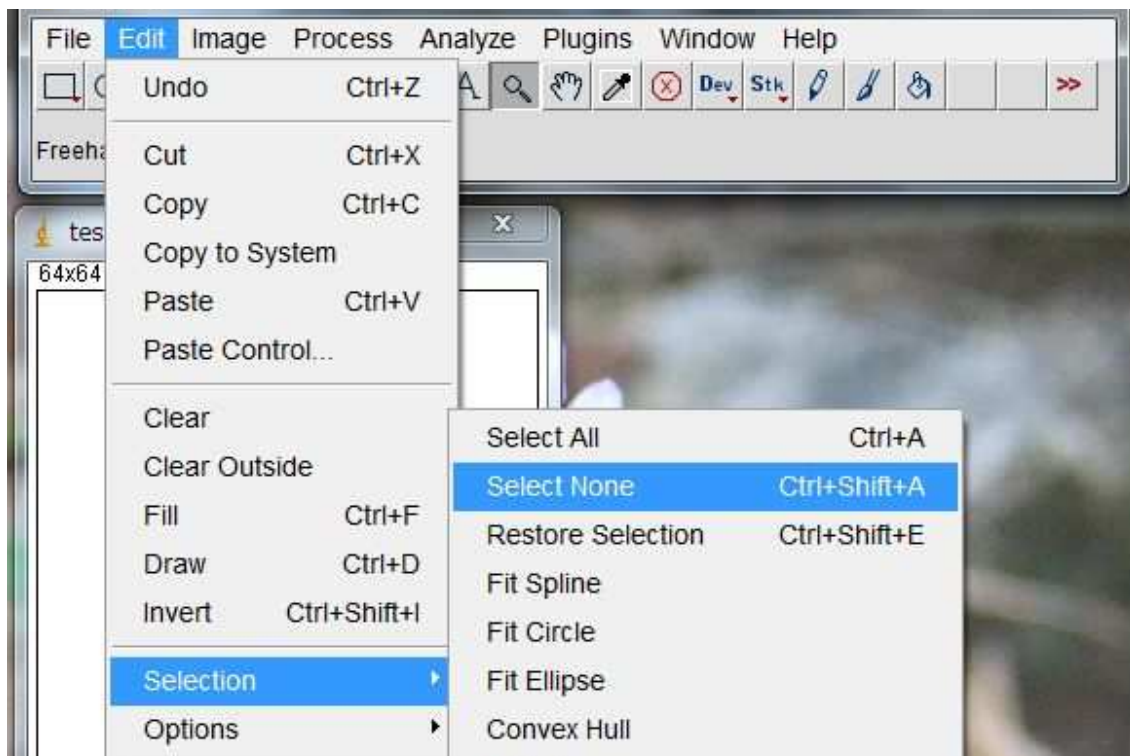


- ④ 選択領域に「Math」→「Add」で数値を加える。(今回は適当に 10 とする)

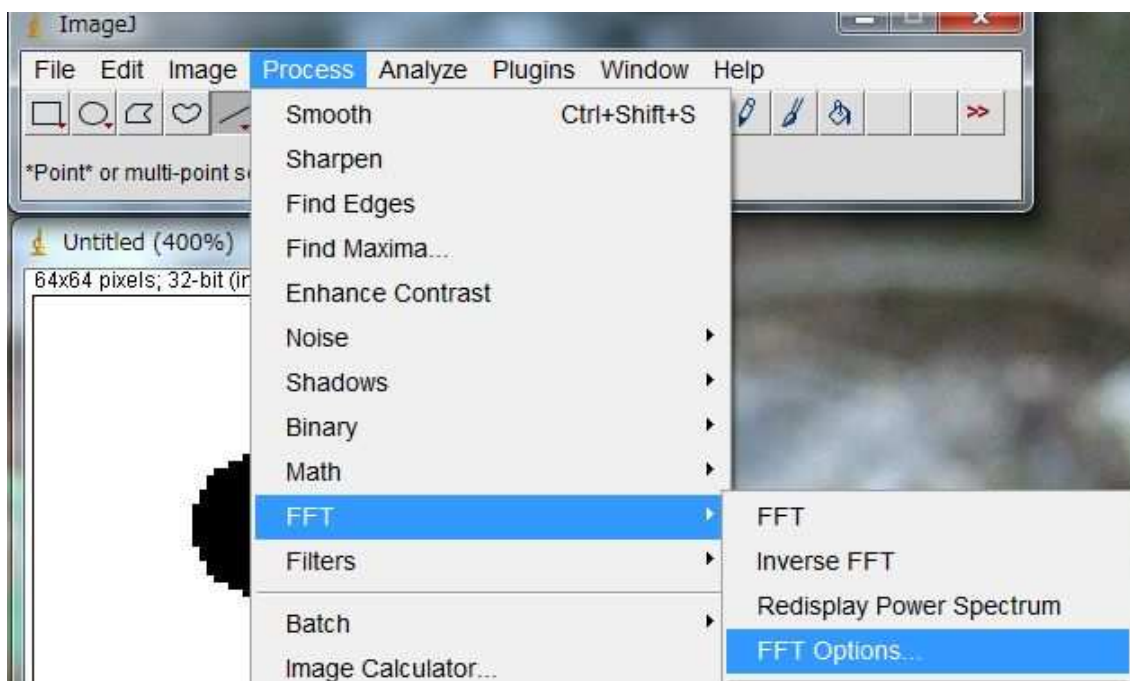




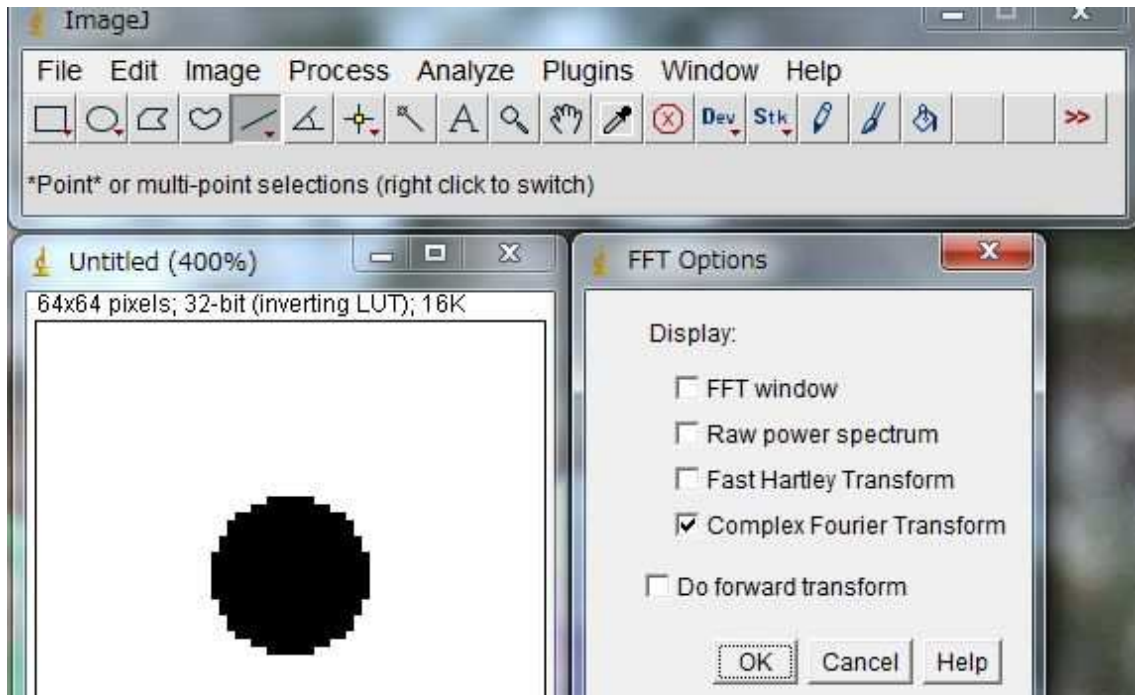
⑤ 選択領域を「Selection」→「Select None」で選択解除する。



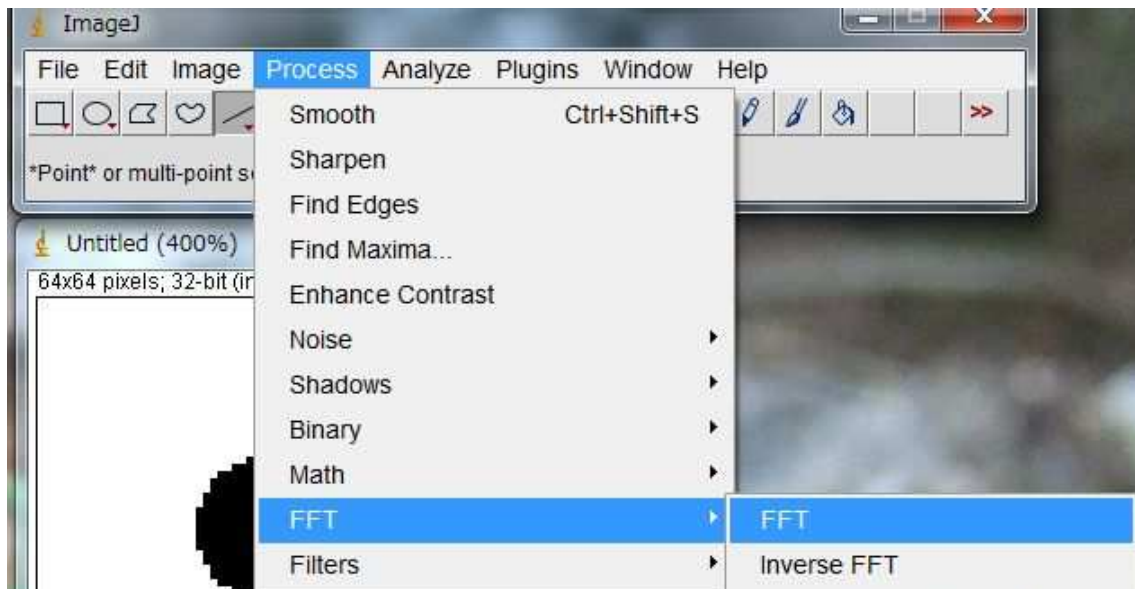
⑥ 「FFT」→「FFT Options」を選択する。



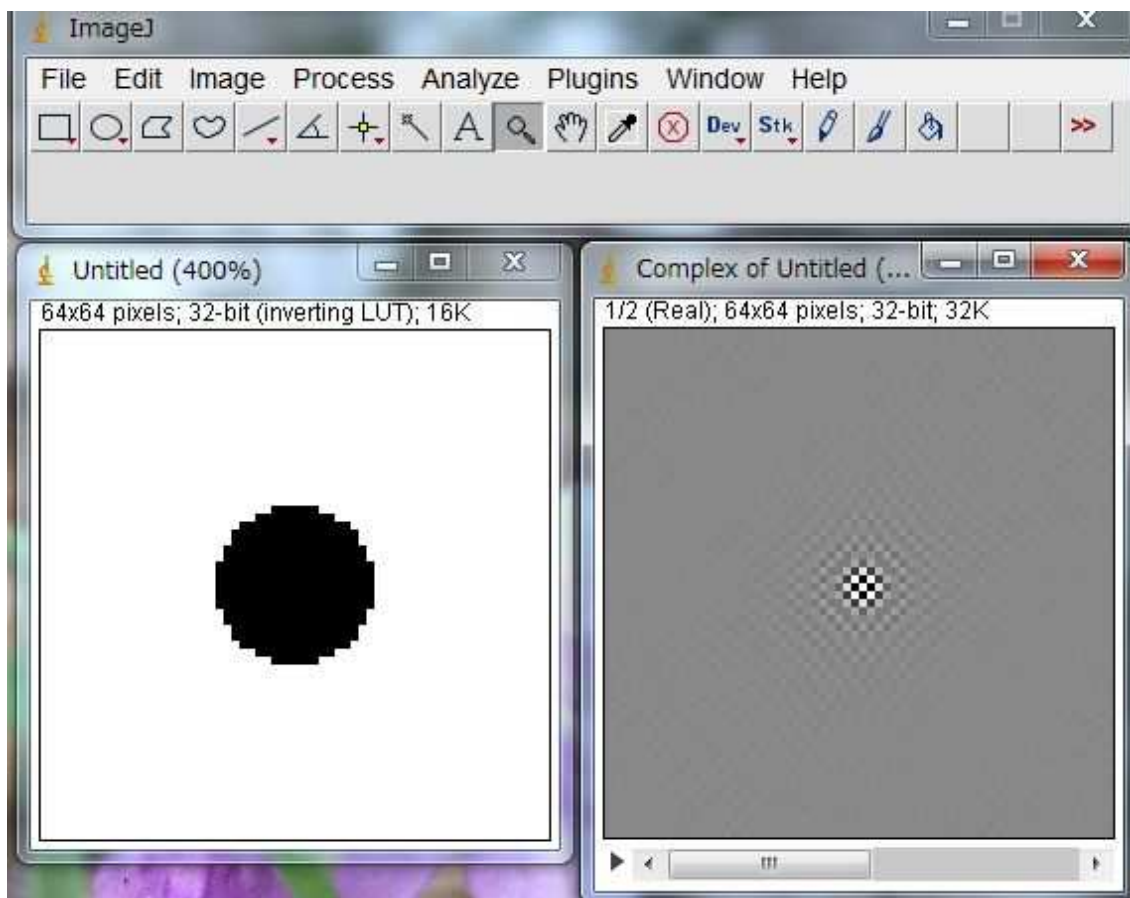
⑦ 「Complex Fourier transform」にチェックする。



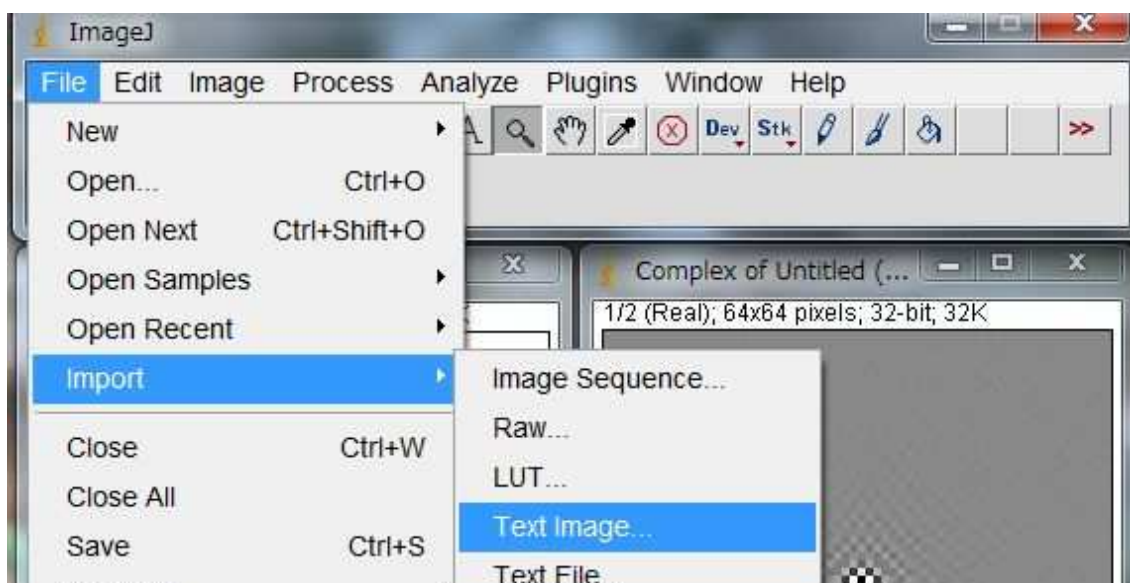
⑧ 矩形波をフーリエ変換する。

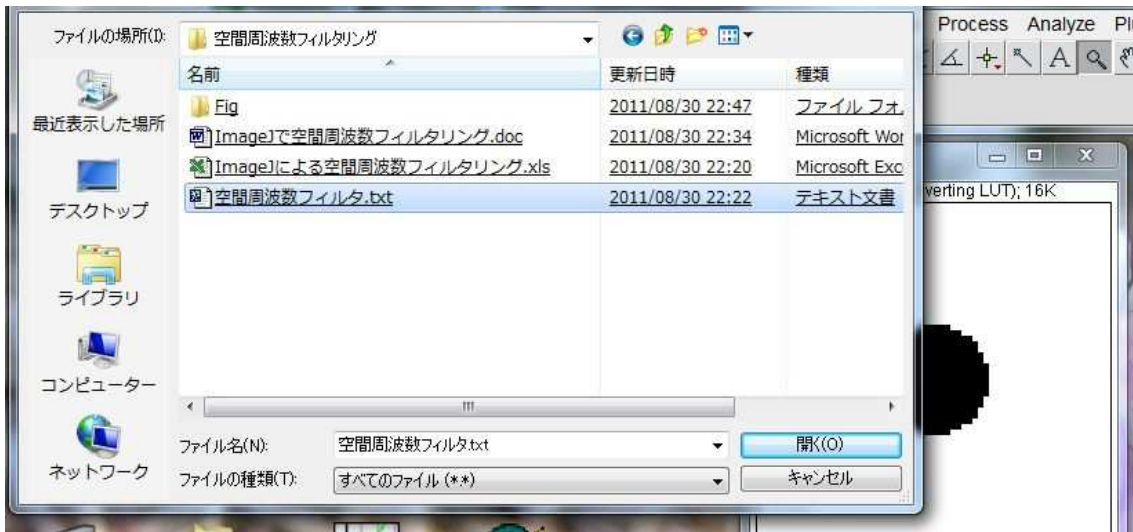


⑨ 右は矩形波をフーリエ変換した結果。

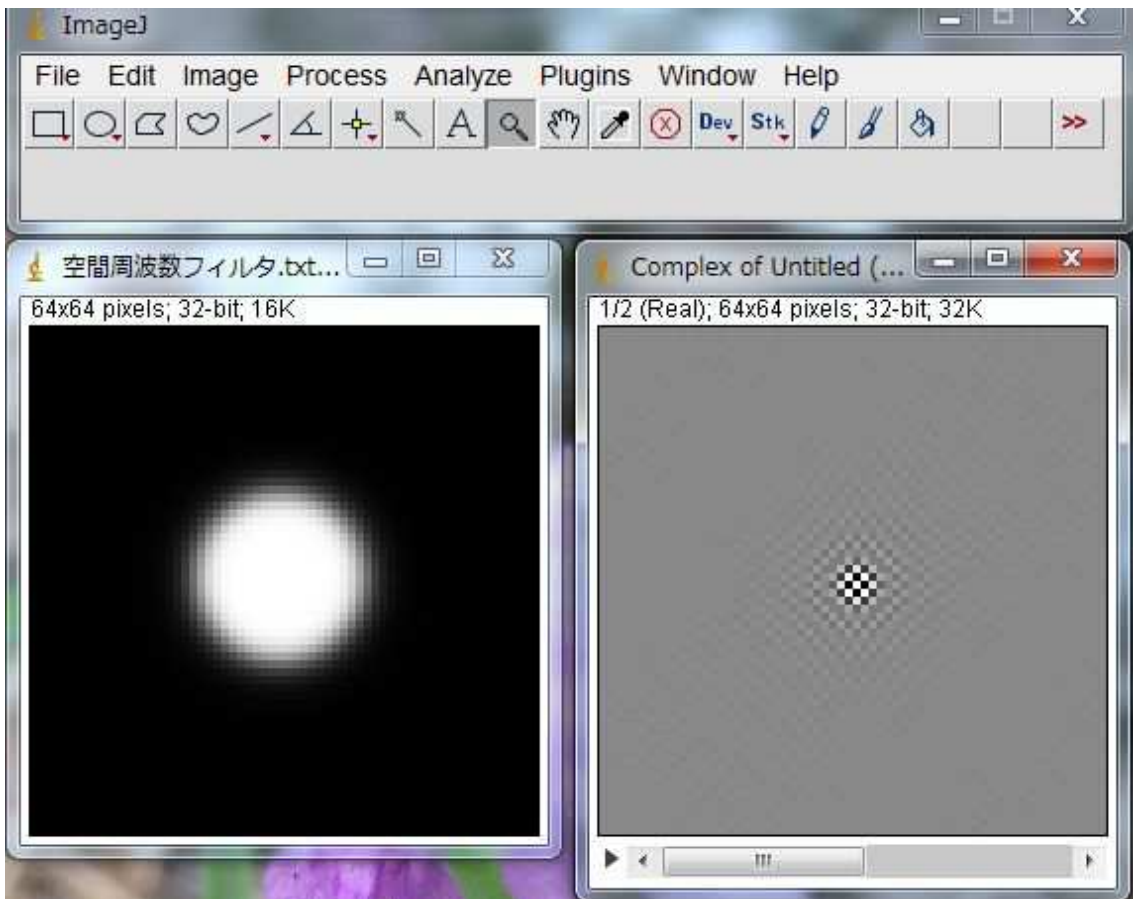


⑩ ここで Excel にて作成したフィルター関数を「Import」→「Text Image」にて読み込む。

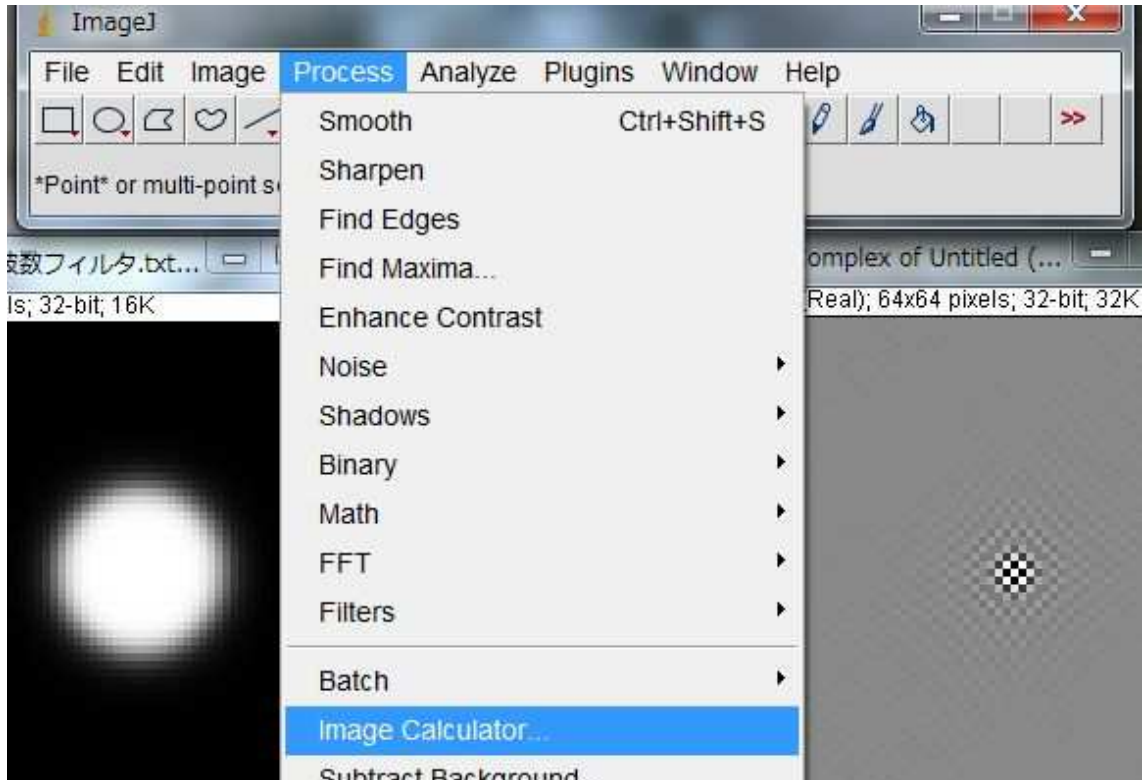




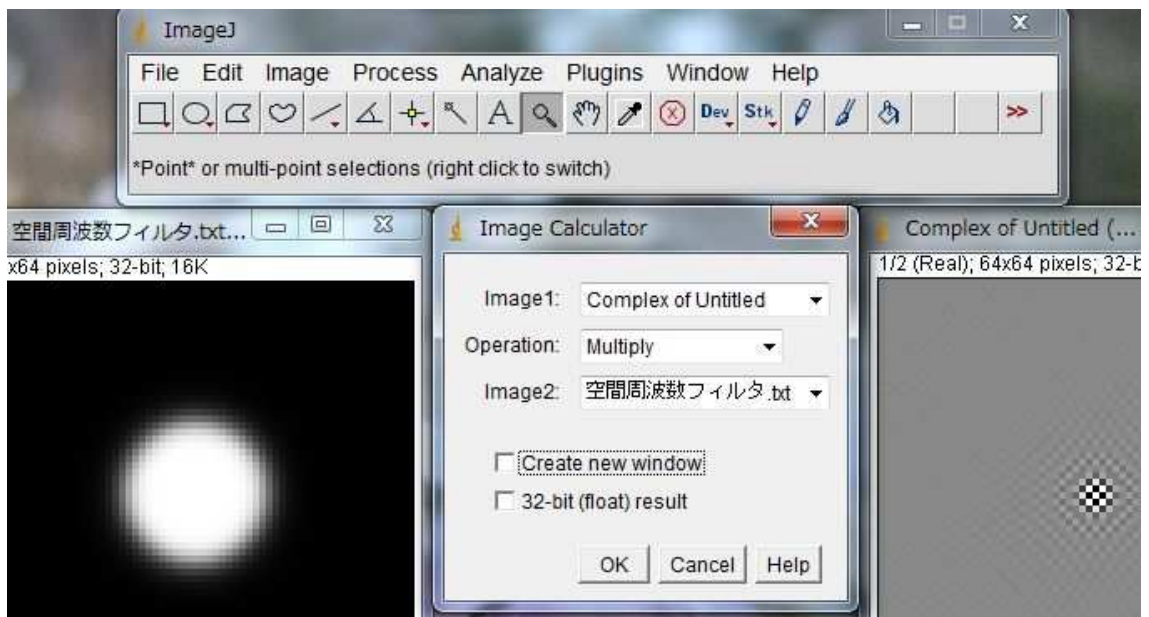
⑪ 読み込んだ空間周波数フィルタ（左）と矩形波のスペクトル（右）



- ⑫ 「Image Calculator」を使って円筒画像のスペクトルと空間周波数フィルタの積を求める。



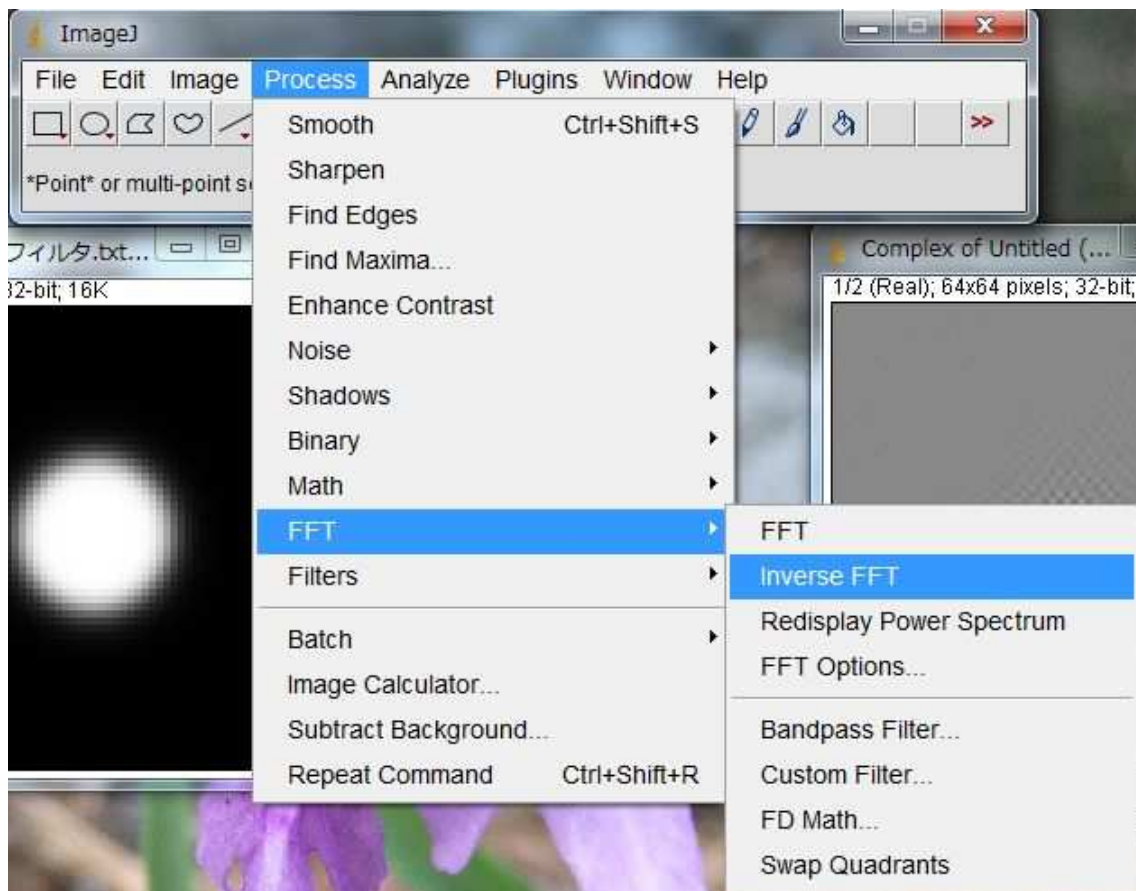
- ※このとき「Create new window」と「32-bit (float) result」のチェックをはずす。



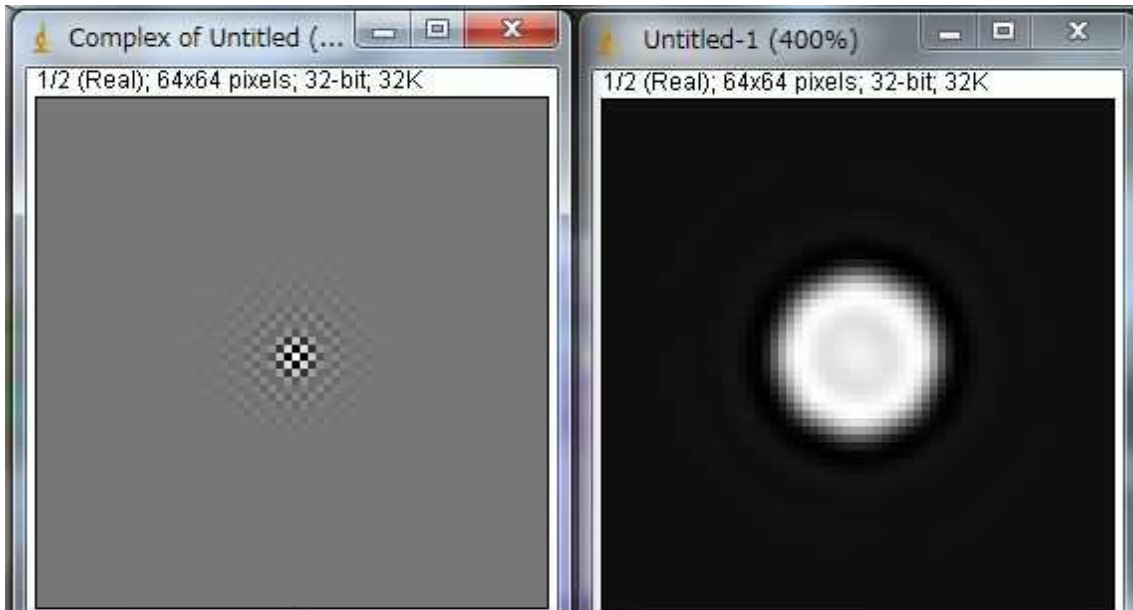
⑬ 「Process Stack?」は「Yes」を選択。



⑭ 結果をフーリエ逆変換する。



⑮空間周波数フィルタリングの完成。



処理結果のラインプロファイル。

