NHF：A Matlab function for edge detection of potential field data using the Normalized Harris Filter

User's manual

Tao Chen
Guibin Zhang
Jun 3, 2020

Introduction

The NHF function is a Matlab script, so it is possible to work with it in any operating system where Matlab is present. In this file, all the details necessary to run, configure and work with NHF are provided. The function is composed of various Matlab functions and three additional sub-functions.

To run NHF.m, you need to provide the following parameters

- Input parameters:

x	required, double, matrix	coordinates of the data along x (north) direction
y	required, double, matrix	coordinates of the data along y (east) direction
v	required, double, matrix	potential field data
relTol	required, double, scalar	relative threshold, [0-1]
k	required, double, scalar	empirical constant, [0-1]

Keep in mind, x, y, and v are in matrix form. If you have scattered data, you should grid it before using this function.

***k* and *relTol* determination**

A larger empirical constant *k* is more conducive to edge enhancement. It is recommended to set the value of *k* to 1.0. Don't change it unless you know what you are doing.

The selection of *relTol* is related to the noise level in the data and the complexity of the underground structure. There is no recommended value for all situations, but a proper initial choice is 1.0×10^{-3}. If the data in good quality, a small *relTol* can help us figure out the detailed structure. Otherwise, a larger *relTol* can improve the robustness of NHF to noise.

When the program is finished, we can get the following output

- Output parameters:

NHFS	double, matrix	normalized Harris filter response
R	double, matrix	Harris filter response (without normalization)

upperEnv	double, matrix	upper envelope used for normalization
ind1	double, vector	index of local maxima before thresholding
ind2	double, vector	index of local maxima after thresholding

In general, NHFS is used for further interpretation. If you want to understand the operation details of NHF, outputting other parameters is a good choice.

Please refer to the paper for the principle of the NHF method:
Tao Chen, Guibin Zhang, Edge enhancement of potential field data using normalized Harris filter.

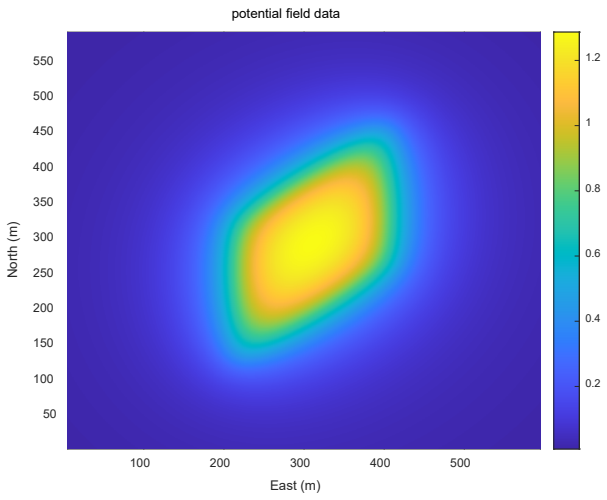
How to run

1) Load the potential field data

```
obs = load('parallelepiped.mat');
obs = obs.obs;
x = obs.x;
y = obs.y;
v = obs.v;
```

we can make a plot for potential field data

```
figure;
pcolor(y, x, v); shading interp; hold on;
xlabel('East (m)');
ylabel('North (m)');
title('potential field data')
colorbar;
```



2) Set the *relTol* and *k*

```
f = 1e-3;      % relTol
k = 1;      % we recommend not to change this.
```

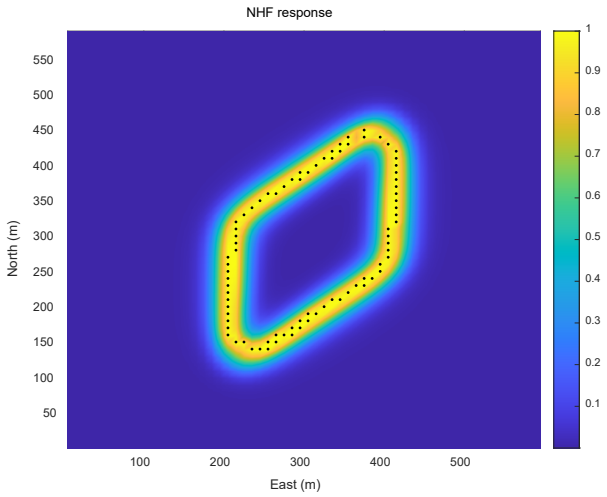
3) Run the NHF.m

```
[NHFR, R, upperEnv, ind1, ind2] = NHF(x, y, v, f, k);
```

You can try different *f* to understand how it changes the result.

4) We can plot the NHF response to obatin the edges of the source body.

```
figure;
pcolor(y, x, NHFR); shading interp; hold on;
xlabel('East (m)');
ylabel('North (m)');
title('NHF response')
colorbar;
```



Refer to the script ‘example.m’ for details.

Contact

Any question about NHF can be directed to any of the authors:

- **Tao Chen:** Department of Earth and Space Sciences, Southern University of Science and Technology, Shenzhen, China (chentaosx@hotmail.com)
- **Guibin Zhang:** School of Geophysics and Information Technology, China University of Geosciences (Beijing), Beijing, China (gbzhang@cugb.edu.cn)