Tool for Analysis of Oscillatory Modes
(TAOM)

version 1.0

User Manual

Developers:

Anwar Aldhafeeri (SoMaS), Gary Verth (SoMaS), Viktor Fedun (ACSE)

Plasma Dynamics Group, The University of Sheffield

January 2019


<table>
<thead>
<tr>
<th>Function name</th>
<th>Description of function</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAOM_v1</td>
<td>This is the main function which should be used to analysis of oscillatory modes of input image.</td>
</tr>
<tr>
<td>redblue.m</td>
<td>This code change the colormap from bright blue, white to bright red [1]</td>
</tr>
<tr>
<td>subplot.tight.m</td>
<td>This code is defining margins and wrapping the existing subplot function [1]</td>
</tr>
</tbody>
</table>

Table 1: Description of the tool functions

1 Introduction

The Tool for Analysis of Oscillatory Modes (TAOM) version 1.0 (TAOM v1.0, written in MatLab) is designed to detect and trace the boundary of binary image of sunspots umbra (or other feature for which boundary can be traced) and then calculate the eigenmodes and eigenfunctions of the shape of the input sunspots image with implies a fixed boundary condition using discrete Laplacian in MatLab. The code scans parameter space for eigenvalues and orthogonal eigenvectors that match the boundary conditions for any given cross-sectional shape. Also this code is designed to find the best elliptical (or other shape) approximation of the sunspot and calculate the eigenmodes/eigenfunctions and provides the comparison between the umbra of the sunspots and elliptical membrane. This code work with binary image only.

2 Usage

This manual includes the following parts:

(1) software environment setting;
(2) description of the tool functions;
(3) overall procedure how to use the TAOM;
(4) examples of use.

2.1 Software environment setting

We recommend to use MatLab 2017a or later versions to avoid any incompatible conflicts with software.

2.2 Description of the tool functions

Apart from the redblue.m programme the all other parts of the code were developed by the Authors. Description and usage of redblue.m programme can be found here: [1] (including copyright details).

2.3 Overall procedure how to use the TAOM

Step 1: Input the parameters

1. Input image input_image
2. Input how many modes do you want to observed \( e N \)
3 Input the grid size $h_S$

4 Input $m_0$ which is the value in the following equation:

$$\Delta f(x, y) - m_0^2 f(x, y) = 0.$$ 

5 To make video for modes input $vd$ equal to 1 otherwise 0

5 To save data for observed modes input $da$ equal to 1 otherwise 0

Step 2: Run the main function

The TAOM_v1 function will work only with binary image.

3 Example 1 (sunspot shape)

In this part, we will show how to analyse the oscillatory modes of the shape ex_1.png which is located in (TAOM_v1) folder.

```matlab
input_image = imread(ex_1.jpg)

e_N=51

h_S=500

m_0=0.5

vd=1

ad=1

[V0,omg,V1,omg1]=TAOM_v1(input_image,e_N,h_S,m_0,vd,da)
```

Processes and Results

Step 1 First, scan the binary image and detect the boundary.
Figure 1: Left panel (a) shows observed sunspots. Right panel (b) shows the observed sunspots with the detection the boundary of sunspots in red colour.

**Step 2** Finding the best ellipse approximation for observed shape.

Figure 2: Approximation of ellipse (the dash line).

**Step 3** Compute the eigenvalues and eigenfunctions of each shape.
Figure 3: Left panel on each image shows the eigenfunction of observed sunspots. Right panel on each image shows the eigenfunction of elliptical sunspot.
Step 4 Make TAOM.avi movie for all modes (optional).

Step 4 Save data for the exact shape TAOM_data.mat file for all modes (optional).

4 Example 2 (modelled shape)

We used the figure ex1_1.png which is located in the (TAOM_v1) folder.

input_image = imread(ex1_1.jpg)

e_N=51

h_S=500

m_0=0.5

vd=1

ad=1

[V0,omg,V1,omg1]=TAOM_v1(input_image,e_N,h_S,m_0,vd,da)

Processes and Results

Step 1 First, scan the binary image and detect the boundary.

Figure 4: Left panel (a) shows the input shape. Right panel (b) shows the input shape with the detection the boundary in red colour.

Step 2 Finding the best ellipse approximation for detected shape.
Figure 5: Approximation of ellipse (the dash line).

**Step 3** Compute the eigenvalues and eigenfunctions of each shape.

**Step 4** Make TAOM.avi movie for all modes (optional).

**Step 4** Save data for the exact shape TAOM.data.mat file for all modes (optional).
Figure 6: Left panel on each image shows the eigenfunction of observed sunspots. Right panel on each image shows the eigenfunction of elliptical sunspot.
5 Release notes

TAOM version 1.0 (2020) can analyse oscillatory modes of exact shape of input binary image.

6 Queries

Anwar Aldhafeeri (SoMaS), aaaldhafeeri1@sheffield.ac.uk, aaaldhafeeri@kfu.edu.sa
Viktor Fedun (ACSE), v.fedun@sheffield.ac.uk
Gary Verth (SoMaS), g.verth@sheffield.ac.uk

7 Citation in the literature

We acknowledge the Plasma Dynamics Group at the University of Sheffield and King Faisal University for support provided. The development of the code was also supported by the European Union’s Horizon 2020 research and innovation program under grant agreement No.824135 (SOLARNET).

References

[1] Copyright (c) 2009, Adam Auton All rights reserved
   https://uk.mathworks.com/matlabcentral/fileexchange/25536-red-blue-colormap


   https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/moler/pdes.pdf

[1] Copyright (c) 2016, Nikolay S. All rights reserved.
   https://uk.mathworks.com/matlabcentral/fileexchange/30884-controllable-tight-subplot