User’s Guide

AnIMAGE: A MATLAB-based tool for generating microstrip antennas with complex shapes.

Brief Descripction of AnIMAGE

AnIMAGE is a software tool developed on MATLAB, designed to make microstrip antennas with complex shapes. This gives antennas better performance and more design options. The software's user-friendly interface makes the design process easier and saves time and effort by automating design calculations. It provides the capability to export complex shapes, allowing CST Studio to simulate their electromagnetic behavior. AnIMAGE lets designers create antennas with complex forms, such as fractals or structures made from images. AnIMAGE is a helpful tool for antennas designers since it offers them new ways to make complex microstrip antennas.

System Requeriments:

This software was created in Matlab R2022b, so the system requirements are the same as this one.

- Processor: Any Intel-based Mac with four cores
- RAM: 4 GB minimum, but 8 GB is recommended
- Hard disk space: 20 GB of HDD space for MATLAB only, 30 GB for a typical installation.
- Operating system: macOS 10.15 (Catalina), macOS 10.14 (Mojave), macOS 10.13 (High Sierra)
- Graphics: A GPU with OpenGL 3.3 or later support is recommended for using the GPU acceleration feature.

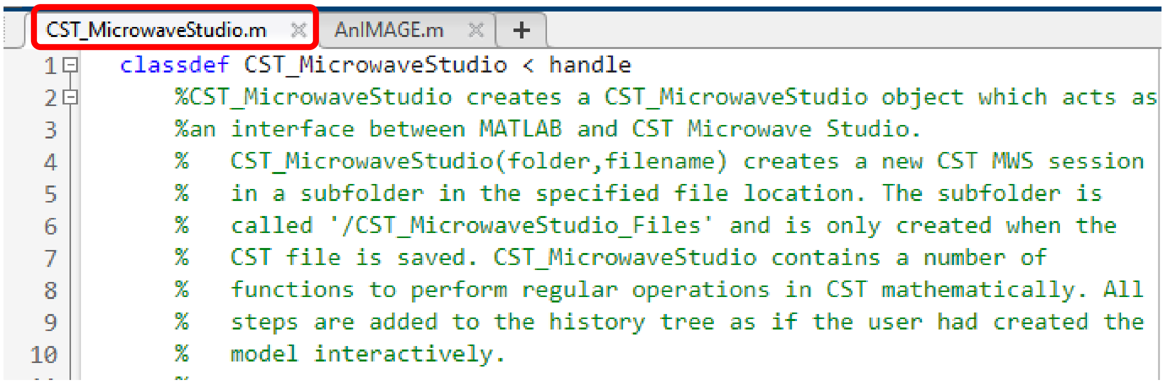
To run the AnImage GUI, it is essential to have the CST_App folder downloaded from the repository, it contains the necessary files that the software use to function properly. Between these, the Application Programming Interface (API) files that connect the CST STUDIO Suite with Matlab extracted via (https://github.com/hgiddenss/CST_App). In addition, as well as other files that enable the users to access in the AnImage GUI from Matlab.

Finally, if you wish to perform an electromagnetic analysis on a structure designed with AnIMAGE, you must have previously installed the CST Studio software on your computer, if it is not installed, the Matlab script “CST MicrowaveStudio.m” will not run correctly

Instructions:

Instructions for installing the software and generating microstrip antennas with complex structures are provided below.

Step 1: Ensure that Matlab and CST STUDIO have been installed on your computer. Then, open Matlab and go to the directory where is contained the downloaded ***CST_MicrowaveStudio.m*** file. Once you have located the file and opened it, select the “Run” button to execute it.
The ***CST_MicrowaveStudio.m*** file is intended to facilitate the integration of CST STUDIO with Matlab R2022b. By executing this file, you can utilize all the CST STUDIO software's features within Matlab.



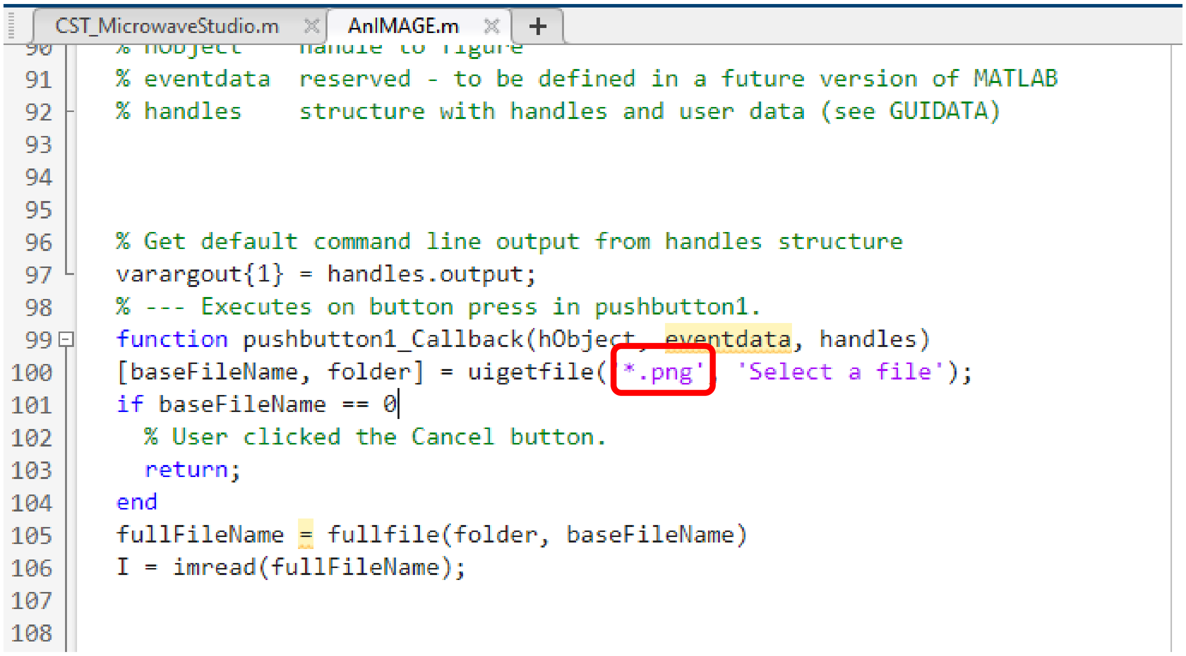
```
classdef CST_MicrowaveStudio < handle
    %CST_MicrowaveStudio creates a CST_MicrowaveStudio object which acts as
    %an interface between MATLAB and CST Microwave Studio.
    %   CST_MicrowaveStudio(folder,filename) creates a new CST MWS session
    %   in a subfolder in the specified file location. The subfolder is
    %   called '/CST_MicrowaveStudio_Files' and is only created when the
    %   CST file is saved. CST_MicrowaveStudio contains a number of
    %   functions to perform regular operations in CST mathematically. All
    %   steps are added to the history tree as if the user had created the
    %   model interactively.
```

Figure 1. *The display of the **CST_MicrowaveStudio.m** file.*

Step 2: Open the ***AnIMAGE.m*** file. The ***AnIMAGE.m*** file contains the code that describes each element of the GUI, such as buttons, menus, text boxes and axes. This code is responsible for the functionality of the software, so it is important to avoid making abrupt changes to the source code unless you have a good understanding of the implications of those changes.

To execute the ***AnIMAGE.m*** file, simply click on the "run" button in the Matlab environment.

Note: The proposed software was configured by default to operate with .PNG images. However, additional image formats such as .JPG, .TIFF among others are supported, if they are not encrypted. As depicted in Figure 2, it could be selected at line 100 of the source code.

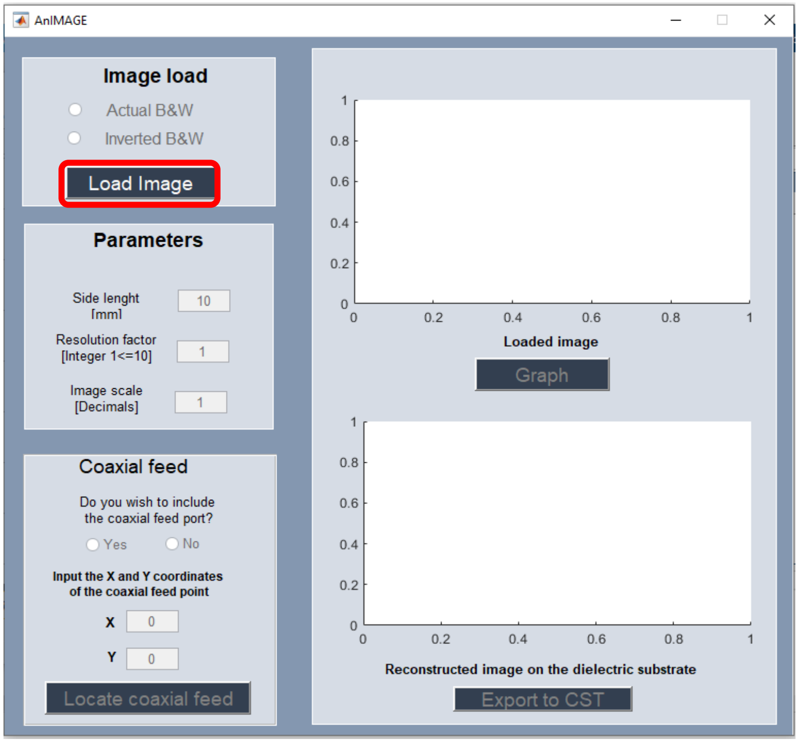


```
 90   % hObject    handle to figure
 91   % eventdata  reserved - to be defined in a future version of MATLAB
 92   % handles    structure with handles and user data (see GUIDATA)
 93
 94
 95
 96   % Get default command line output from handles structure
 97   varargout{1} = handles.output;
 98   % --- Executes on button press in pushbutton1.
 99   function pushbutton1_Callback(hObject, eventdata, handles)
100   [baseFileName, folder] = uigetfile('*.png', 'Select a file');
101   if baseFileName == 0
102     % User clicked the Cancel button.
103     return;
104   end
105   fullFileName = fullfile(folder, baseFileName)
106   I = imread(fullFileName);
107
108
```

Figure 2. *The display of the **AnIMAGE.m** file.*

Step 3: After executing the code contained in the file, the AnIMAGE GUI will start, which you can then use to analyze and process your images. All elements of the GUI will be disabled by default until the desired image to be used is loaded.

The user can import the image by selecting the "load image" button in the upper-left corner of the user interface, as shown in Figure 3.



Side lenght [mm] 10

Resolution factor [Integer 1<=10] 1

Image scale [Decimals] 1

X 0

Y 0

Figure 3. *User interface of AnIMAGE. The red box highlights the position of the **load image** button*

Step 4: After navigating to the folder containing the image to be uploaded, the user must select it in .png format. To obtain a better fill in the construction of the figure, it is recommended to use a previously binarized or grayscale image.

AnIMAGE provides the option to automatically binarize an image if the user does not already possess one. This signifies that the software will convert the image into a two-tone format so that it can be processed correctly.

In fact, the user interface supports two distinct forms of image reconstruction, as illustrated in the following examples:

Example: Reconstruction using previously binarized image

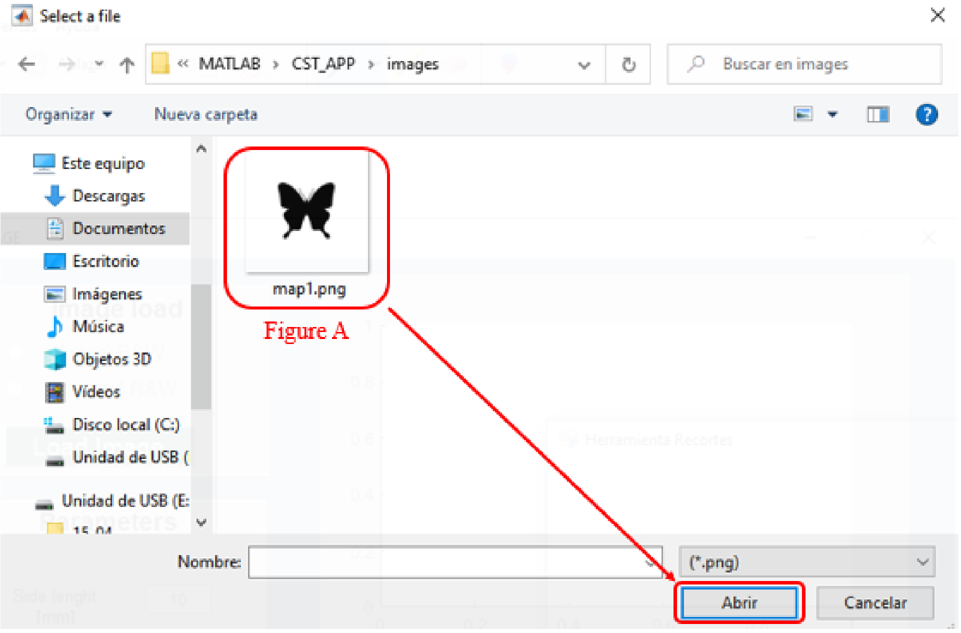


***Figure 4**. Select and load the image (Figure A). In this case, an image of a butterfly was proposed since it has a complex symmetrical shape.*

Step 1: Once Figure A has been loaded, we have two options: Actual B&W or Inverted B&W. The primary distinction between these two options is the area that will be filled with copper. In fact, the white color indicates the cooper region, and the black color the substrate.

Option 1: Actual B&W

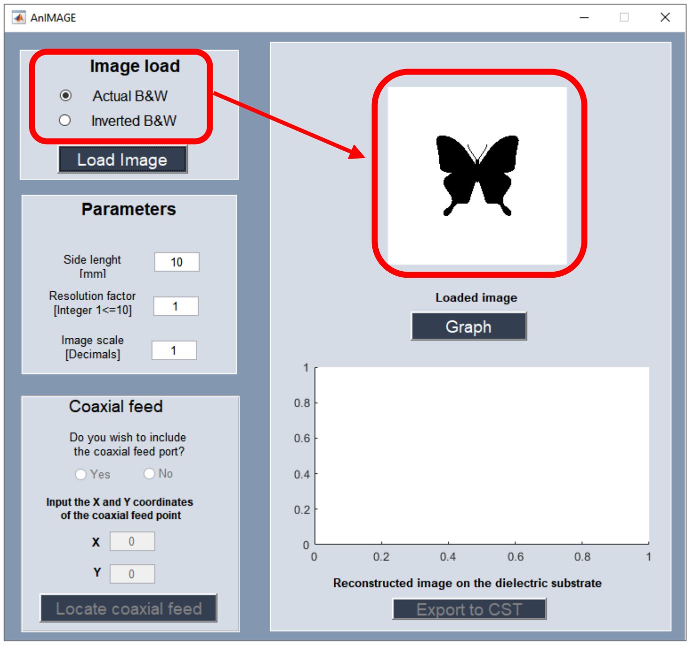
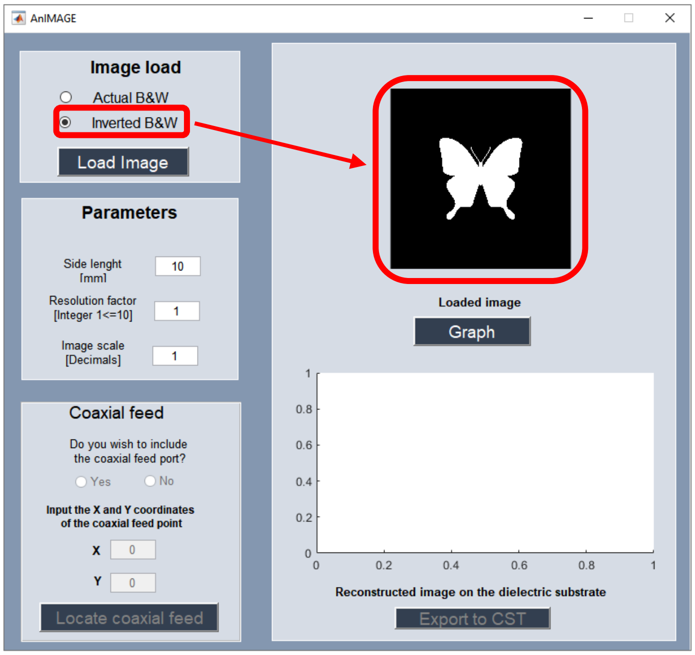


***Figure 5.** Select* Actual B&W option. The black color indicates the cooper region.

Option 2: Inverted B&W



○ Actual B&W
● Inverted B&W

Side lenght [mm] 10
Resolution factor [Integer 1<=10] 1
Image scale [Decimals] 1

Do you wish to include the coaxial feed port?
○ Yes ○ No

Input the X and Y coordinates of the coaxial feed point
X 0
Y 0

Figure 6. *Select Inverted* B&W option. The black color indicates the cooper region.

Step 2: Now, the parameter tab will be enabled. Then, the user can select the optimal parameters for reconstructing the image on a dielectric substrate. Options that can be configured include the following:

a. **Side length:** The dimension of the substrate in millimeters must be selected here. In this instance, the two dimensions of the substrate (length and width) will be the same to prevent deformation of the previously loaded image. As shown in Figure 7, for instance, we can input 30 mm for this parameter.
b. **Resolution factor:** This parameter accepts values from 1 to 10. Since it reconstructs pixel by pixel, the optimal resolution for image reconstruction is a factor of 1. However, this significantly increases the reconstruction time. In this example, a factor of 2 is utilized, indicating that a Image reconstruction will be done every 2 pixels.
c. **Image Scale:** This parameter allows adjusting the size of the image so that it is proportional to the substrate. In this example, we will use a scaling factor of 1.5 as illustrated in Figure 7.
d. After entering the parameters, pick the **Graph** button.
e. In the box to the right, the image will appear as shown, with the blue area representing the copper, the white portion representing the substrate, and the axis showing the dimensions of the substrate (in this case, 30mm x 30 mm).

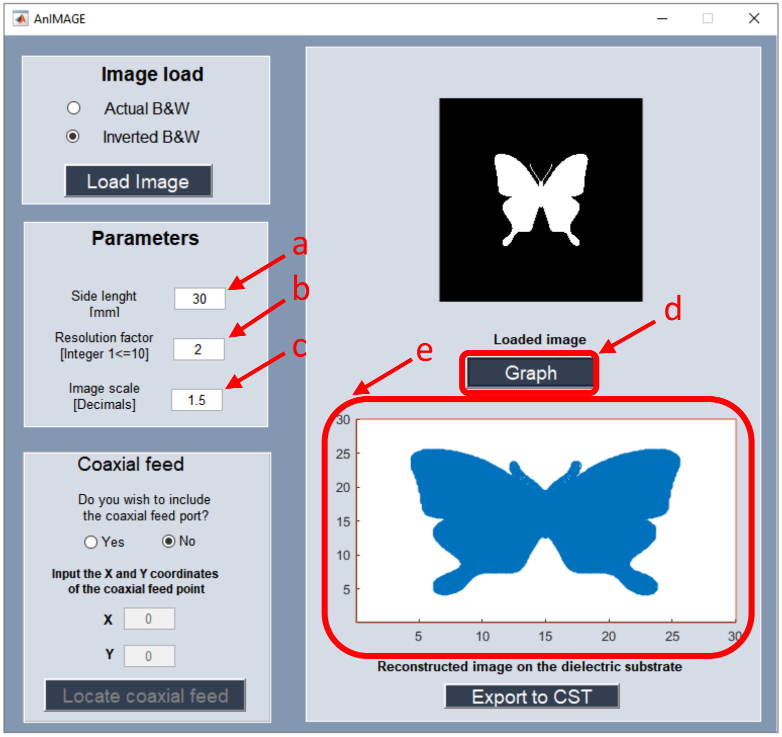


Image load: ○ Actual B&W ● Inverted B&W

Side lenght [mm]: 30

Resolution factor [Integer 1<=10]: 2

Image scale [Decimals]: 1.5

Do you wish to include the coaxial feed port? ○ Yes ● No

Input the X and Y coordinates of the coaxial feed point

X: 0

Y: 0

Figure 7. *Input of substrate parameters and showing of the reconstructed image based on the entered parameters.*

Figure 8 illustrates how the reconstruction alters when the image scale is increased to 5, while all other parameters remain unchanged. The butterfly image now exceeds the size of the defined substrate, as shown in Figure 8. It is essential to clarify that this could be exported to the CST STUDIO software in the same manner without issue.

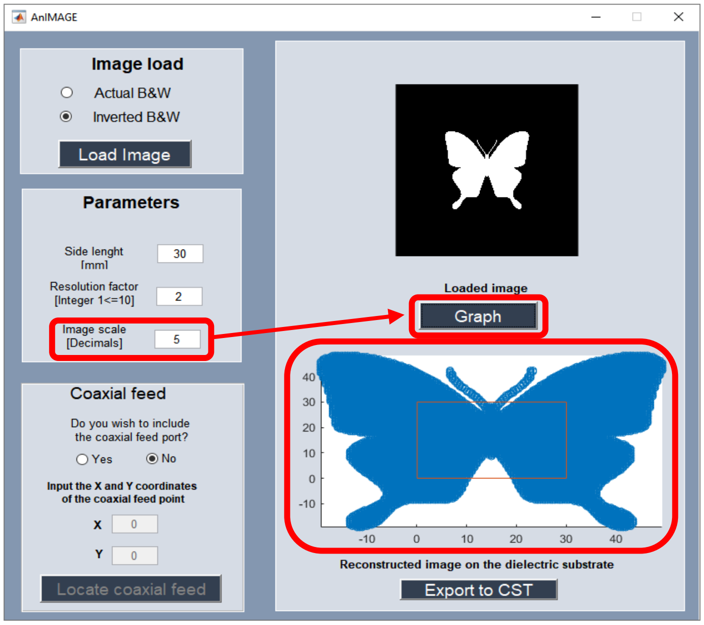


Image load: ○ Actual B&W ● Inverted B&W

Side lenght [mm]: 30

Resolution factor [Integer 1<=10]: 2

Image scale [Decimals]: 5

Do you wish to include the coaxial feed port? ○ Yes ● No

Input the X and Y coordinates of the coaxial feed point

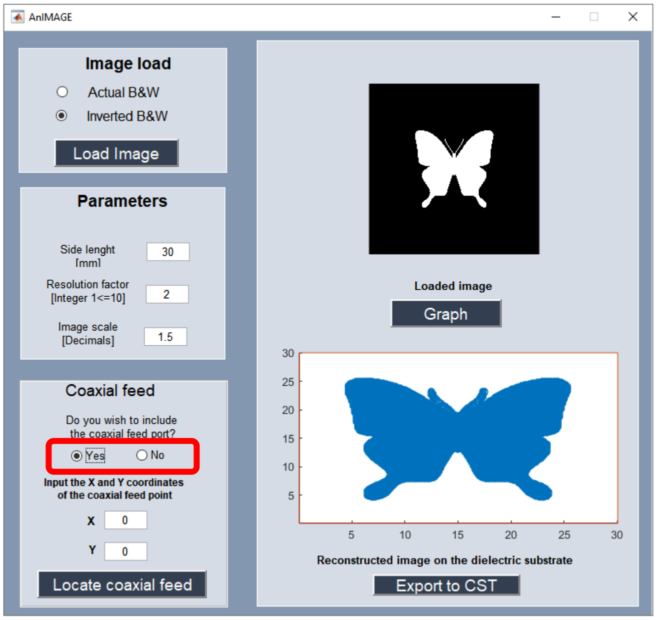
X: 0

Y: 0

Figure 8. *Variation of the image scale from 1.5 to 5, while the other parameters remain unchanged.*

Step 3: Now, the Coaxial feed option is available. Then, the user has the option of adding the coaxial cable. In the proposed example, we choose the option yes, as shown in Figure 9.

Note: the impedance of this port depends on the position where it is placed.



○ Actual B&W
● Inverted B&W

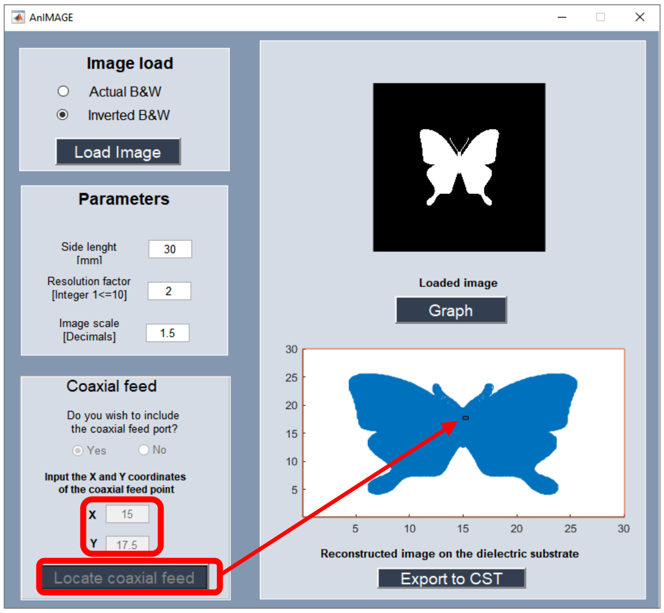
Side lenght [mm]: 30
Resolution factor [Integer 1<=10]: 2
Image scale [Decimals]: 1.5

Do you wish to include the coaxial feed port?
● Yes ○ No

Input the X and Y coordinates of the coaxial feed point
X: 0
Y: 0

Figura 9. The user decides whether to add the coaxial feed to the designed antenna.

Step 4: Next, AnIMAGE requests the X and Y coordinates in millimeters where you want to place it. For example, we input X=16 mm and Y=17.5 mm as is depicted in Figure 10. After that, click again on the **Graph** button.



○ Actual B&W
● Inverted B&W

Side lenght [mm]: 30
Resolution factor [Integer 1<=10]: 2
Image scale [Decimals]: 1.5

Do you wish to include the coaxial feed port?
● Yes ○ No

Input the X and Y coordinates of the coaxial feed point
X: 15
Y: 17.5

Figure 10. Input the coaxial feed's coordinates in the GUI.

Step 5: Then, press the **Export to CST** button as is depicted in Figure 11, a new working window will open in CST Studio and start building the figure pixel by pixel according to the chosen Resolution factor and the image scale.

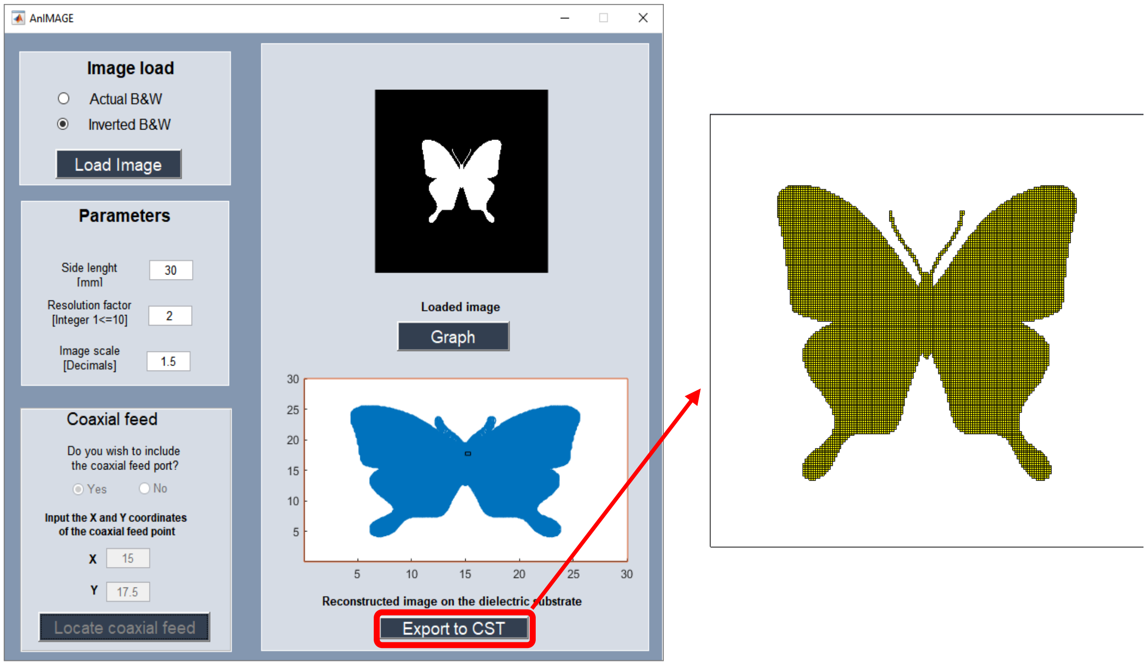


Image load

○ Actual B&W
● Inverted B&W

Parameters

Side lenght [mm]: 30
Resolution factor [Integer 1<=10]: 2
Image scale [Decimals]: 1.5

Coaxial feed

Do you wish to include the coaxial feed port?
○ Yes ○ No

Input the X and Y coordinates of the coaxial feed point
X: 15
Y: 17.5

Figure 11*. Reconstruction of the figure in CST Studio based on the chosen parameters.*

Step 6: Once the image is complete, the pixels will be automatically joined to reconstruct the figure. The following components are identifiable in Figure 12.

a. Substrate. The system is constructed with a FR-4 substrate by default; however, this can be changed in CST STUDIO. The white color indicates the presence of the substrate, and its thickness is 1.6 mm by default also.
b. Radiant patch (Copper). The yellow color indicates the presence of the cooper layer. The thickness is 35 µm by default.
c. Aggregated Coaxial feed (optional).
d. Ground (Copper). The yellow color indicates the presence of the cooper layer. The thickness is 35 µm by default.

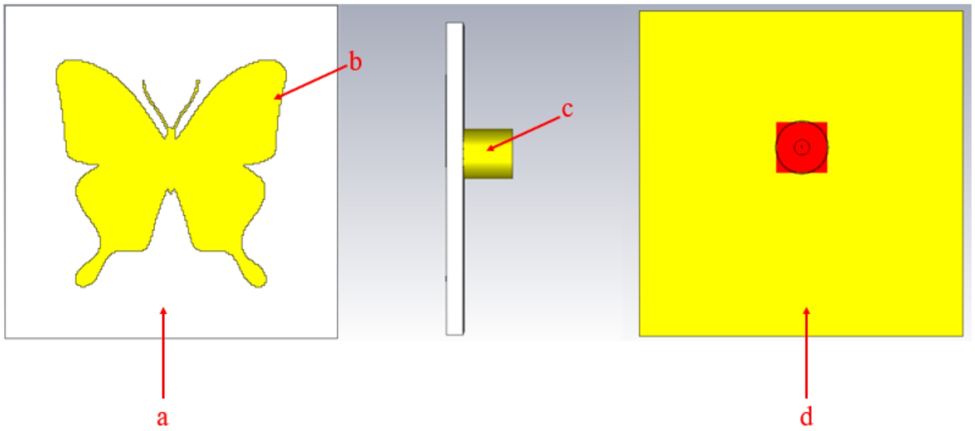


Figure 12. *Antenna built in CST. The different components of the designed antenna are shown here.*

Step 7: Finally, the user can simulate the electromagnetic behavior of the designed antenna using CST STUDIO.

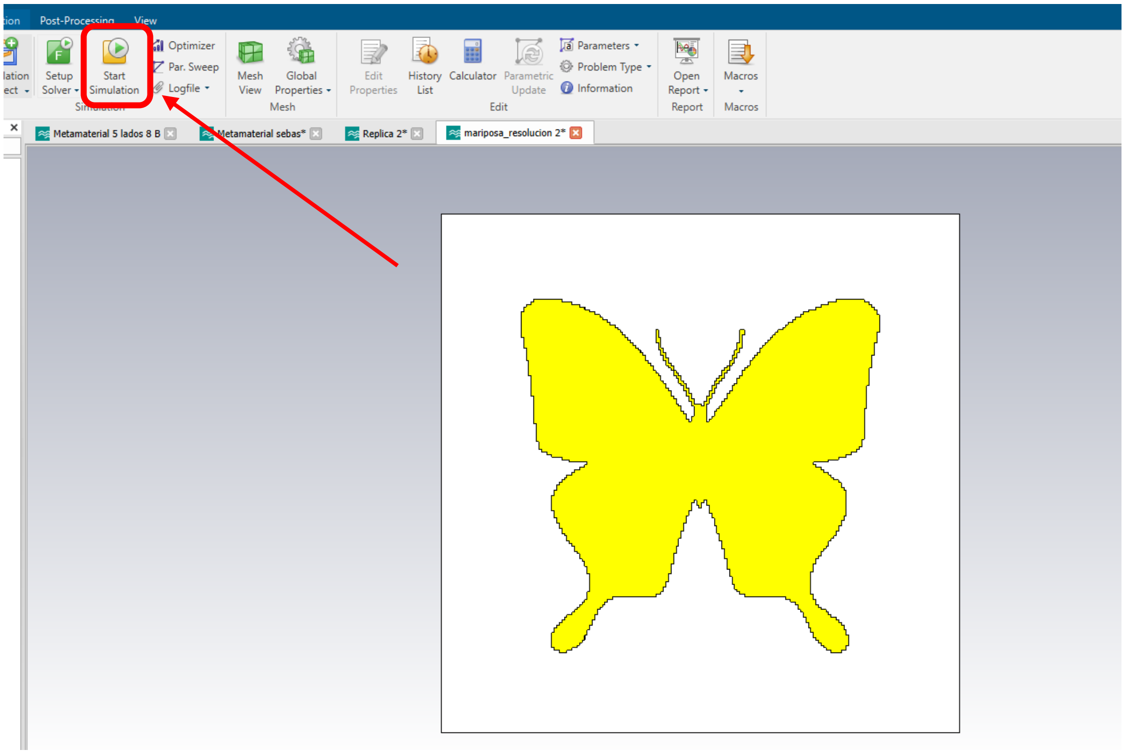


Figure 13. *The antenna is ready to be simulated in CST STUDIO, to know its electrical characteristics.*