GiNA. Instructional demo

Diaz-Garcia Luis Antonio.

March 15, 2016

Laboratory of Cranberry Genetics and Genomics, Horticulture Department. University of Wisconsin-Madison, Madison, WI.

- What is GiNA?

GiNA is free software for measuring horticultural traits related with the shape- and color of fruits, vegetables, and seeds. The software was written in both $\mathbb{R}$ and MATLAB programming languages, it uses conventional images from digital cameras and process up to 11 different morphological traits such as length, width, two-dimensional area, volume, projected skin, surface area, RGB color, among other parameters.

- What do I need to use GiNA?

A digital camera. It can be an economic one, but a DSRL is prefered. A tripod or camera holder. A contrasting background.

- Where can I get GiNA?

The functions are deposited in the Cranberry Genetics and Genomics Lab website (http://cggl.horticulture.wisc.edu/home-page/); the version written in $\mathbb{R}$ is also available in the Comprehensive $\mathbb{R}$ Archive Network (CRAN).

- Can I citate GiNA?

Yes. Here is the information:

Diaz-Garcia L., Covarrubias-Pazaran G., Schlautman B., Zalapa J. 2016. GiNA, an Efficient and High-Throughput Software for Horticultural Phenotyping. *In revision.* PlosONE.

DEMO

The next demo is intended to show the details, requierments, and output, of each step when using GiNA for measuring objects in a picture. This demo uses GiNA for MATLAB.

1 Picture recording

GiNA is intended to be a software for processing multiple pictures efficiently and provide an highly-accesible database with all the phenotypic data. Before taking pictures, it is necesary to spend some time optimizing different light conditions and camera settings such as light sensitivity, zoom, shutter speed, and aperture. In addition, avoiding camera flash reduces light reflection in objects in the image, which appear as bright spots. Finally, users should ensure that the camera is aligned with the center of the background and is properly focused on the fruits. As control, references on each side of the background must be included to allow data normalization. The references must be located in the sides of the picture. Good results have been seen using 6 circular references (about 1 inch diameter) on each side of the picture. If you are going to analyze just one picture, you can hold the camera with your hands and take the picture, however, if multiple photos are going to be taken, you must used a firm and stable camera holder such as a tripod or a device like the shown in Figure 1.

Usually, 2MB is enough resolution for obtaining good results from GiNA.

It is important to keep track of the names assigned to the photos, since GiNA will use those names for processing the final output.

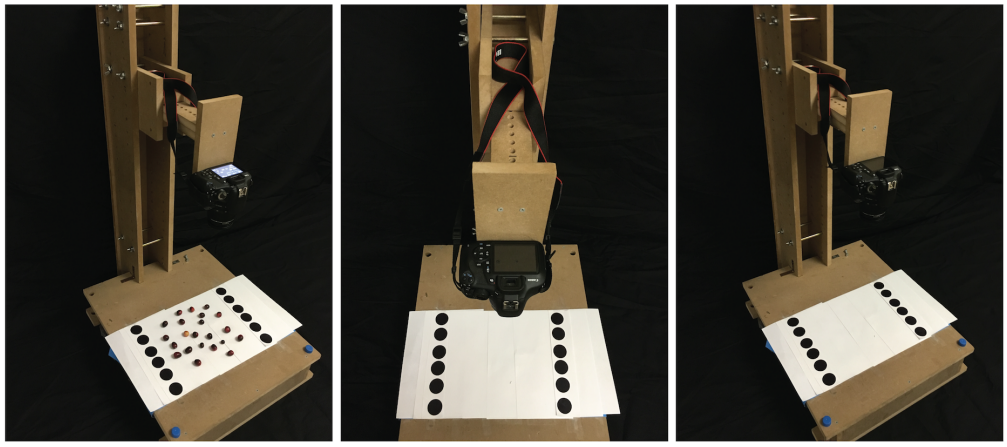


Figure 1: Example of a home-made camera holder. The arm holding the camera is adjustable, it allows to move the camera in both vertical and horizontal directions.

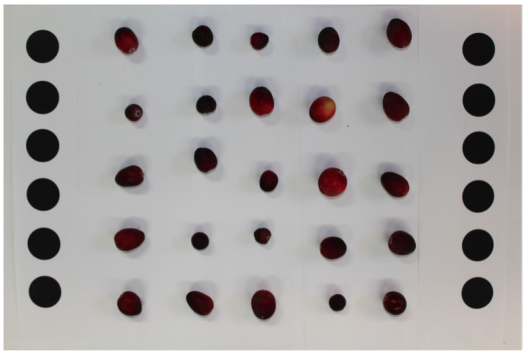
2 Setting GiNA's parameters

2.1 Arguments

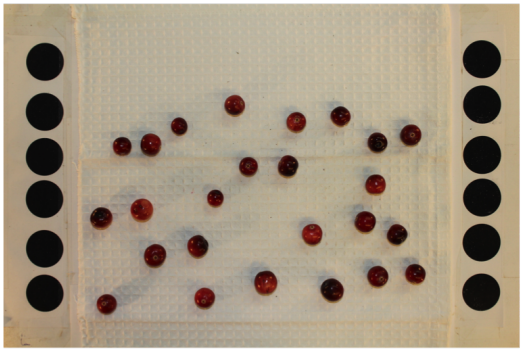
Once the pictues are taken, it is important to set the best parameters for operating GiNA.

These are the arguments:

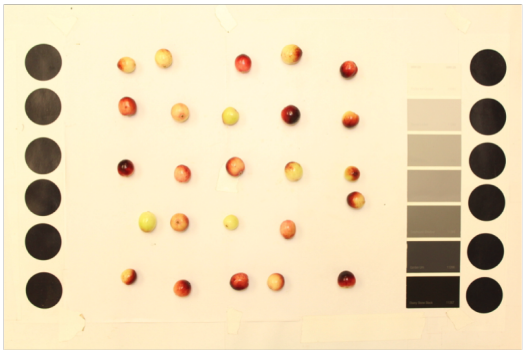
```
>>data=GiNA(folder,resizeIndex,method,minApx,realLength,keepOrder,showFig,writeT)
%% if you just one to try GiNA, you can start with this:
>>data=GiNA('/yourPicture.JPG',0.5,[100,3,6,6],600,2.54,false,true,true) % This means that ...
    GiNA will analyze the picture 'yourPicture.JPG', which will be resized to 1/4 of the ...
    original size and the background extraction will be done using a threshold (100) in ...
    channel 3 (blue). The number of references on each side of the picture are 6 circular ...
    objects with diameter=2.54cm (1 inch). Objects with an area (in pixels) less than 600 ...
    wont be considered, GiNA wont be tracking the order of the fruits, the the program ...
    will produce a couple a couple of figues showing the results and a .csv file will be ...
    written.
%%
%% Here are the details:
%%
>>data=GiNA(folder,resizeIndex) % where 'resizeIndex' indicates the reduction factor for ...
    resizing the image. High-resolution images (more than 4MB) can take more time for ...
    processing. DEFAULT = 0.5.
>>data=GiNA(folder,resizeIndex,method) % where 'method' indicates the methodology to use ...
    in the background extraction step. There are two options: predefined threshold or ...
    neural network approach. For the first option, 'method' has to be a vector containing ...
    (i) a threshold (1-255), (ii) the channel in which the threshold will be applied (R ...
    (1), G (2) or B (3)), (iii) number of references in the left side of the picture, and ...
    (iv), the number of references in the right side of the picture. When the neural ...
    network approach is used, 'method' has to be a structure containig two fields: network ...
    and refs. The first one should content the neural network generated by netGiNA (see ...
    details bellow), the second field has to be a vector of length=2 containing the number ...
    of references in each side of the picture. DEFAULT = [150 3 1 1] (threshold-based ...
    approach).
>>data=GiNA(folder,resizeIndex,method,minApx) % where 'minApx' indicates the minimum area ...
    (in pixels) of objects to be considered. DEFAULT = 500.
>>data=GiNA(folder,resizeIndex,method,minApx,realLength) % where 'realLength' is the ...
    length of the references in a specifc unit such as cm or inches. This number will be ...
    used for transform measurments (origianlly produced in pixels) into a desired unit. ...
    When this argument is defined, addtional data fields with adjusted measurments (using ...
    the specific units) are produced (see detailes bellow). DEFAULT=FALSE. When used, it ...
    has to be a number.
>>data=GiNA(folder,resizeIndex,method,minApx,realLength,keepOrder) % where 'keepOrder' is ...
    a logical indicator to keep track of the objects presented in the figure. When ...
    'keepOrder' is TRUE, the picture must contain the objects arranged in a square grid. ...
```



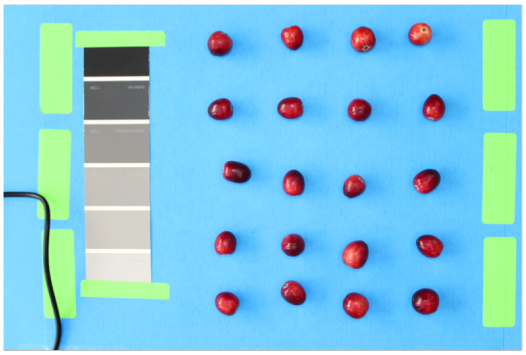
- The picture is centered.
- Good contrast between fruits and background.
- This arrange is perfect when the argument 'keepOrder' is used.
- There are not bright spots in the fruits.



- There are some shadows but it is ok.
- The fruits don't have to be arranged in an specific way (just if 'keepOrder' is not used).



- Too much light
- There is no contrast between yellow fruits and the background



- A bad picture.
- A chord is in the left.
- Although different shapes can used as references, the color scale in the left is mixed with the same color than the references.
- There are some bright spots in the fruits.

Figure 2: Examples of good and bad pictures.

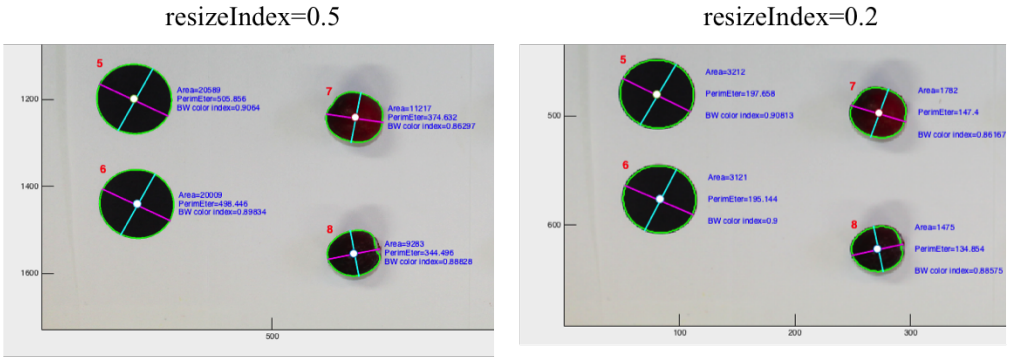


Figure 3: Resizing a photo affects the definition of recognized objects.

```
        If 25 fruits will be analyzed, the arrange could be 5x5. DEFAULT = FALSE.
12  >>data=GiNA(folder,resizeIndex,method,minApx,realLength,keepOrder,showFig) % where ...
        'showFig' is a logical indicator to produce figures. When 'folder' is a directory with ...
        multiple pictures, 'showFig' will be always FALSE. DEFAULT=FALSE.
13  >>data=GiNA(folder,resizeIndex,method,minApx,realLength,keepOrder,showFig,writeT) % where ...
        'writeT' is a logical indicator to write the results in an .csv file. DEFAULT=FALSE.
```

2.1.1 Resize index

Usually, photos with a resolution of 2MB are enough for getting good results from GiNA. If you took pictures with a bigger resolution, you can resize them using the argument 'resizeIndex' as follows:

```
1  >>resizedPhoto=GiNA('/Desktop/pic1.JPG',0.5,[100 3 6 6],700,2.54,false,true,false)  % ...
       resizefactor = 0.5
2  >>originalPhoto=GiNA('/Desktop/pic1.JPG',0.2,[100 3 6 6],700,2.54,false,true,false)  % ...
       resizefactor = 0.2
```

2.1.2 Background extraction method

Depending on the color contrast between object (e.g., fruit) and background, two strategies can be applied for segmentation. When there is a very marked contrast between the object and the picture background, a threshold in one or multiple RGB channels can be set as cutoff value for removing the background from the picture. Commonly, darker (on a white background) or lighter (on a dark background) objects with small color variation can be analyzed by using threshold-based segmentation. When there is significant variation in the color of the object (e.g. one side of the object is dark red and the other side is yellow), or there is not evident contrast between the objects and background, a neural network approach can be implemented. This strategy requires an extra set of images including one picture of the color and texture variation in the background (without the references), and another from the objects. The neural network approach can be used in all the same situations that the threshold-segmentation strategy is appropriate.

Here is an example of background extraction setting a threshold in the channel 3:

```
1  >>data=GiNA('/Desktop/pic1.JPG',0.3,[100 3 6 6],700,2.54,false,true,false)  % setting 100 ...
       in channel 3 (blue)
2  >>data=GiNA('/Desktop/pic1.JPG',0.3,[180 3 6 6],700,2.54,false,true,false)  % setting 180 ...
       in channel 3 (blue)
```

As you can see in Figure X, defining an approaite threshold is probably the most important argument in GiNA's function. An excessive threshold will extract the background incorrectly.

If the objects that you can to measure with GiNA have a color gradient and it hard to select and approiate contrasting background, the neural network approach is a better option. In this case, you have to run the

threshold 100 in channel 3 threshold 180 in channel 3

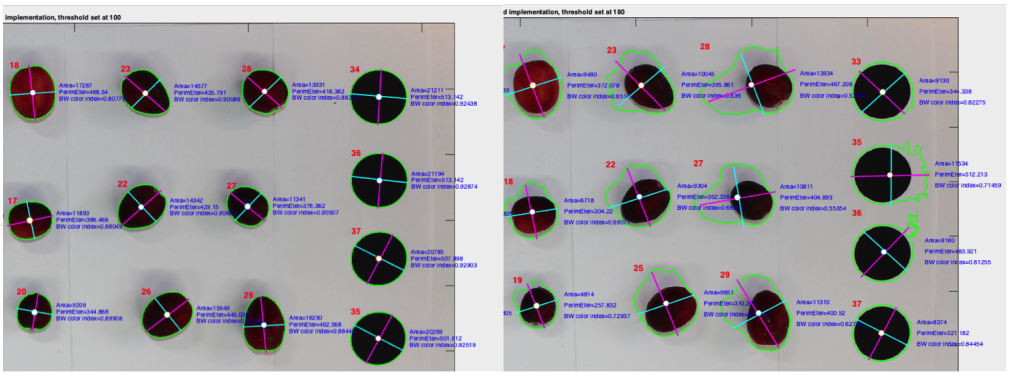


Figure 4: Setting the threshold for background extraction.

function `netGiNA`. This function is very simple, it takes a picture of each of the most overepresented colors in the picture (you have to provide the pictures), and build a network that will be used in `GiNA` function. Here is an example:

```
network=netGiNA('/Desktop/background.JPG','/Desktop/fruit1.JPG','/Desktop/fruit2')
% the first arguments has to be the background, after that, multiple pictures representing ...
    the fruit colors can be added.
method.network=network
% the network object that netGiNA produces has to be added to a structure (the field name ...
    has to be 'network'). In another field called 'refs', the number of references on each ...
    side of the picture must be specified (starting from the left).
method.refs=[6 6]
% the structure is going to be the argument 'method' in the function GiNA. (It is going to ...
    replace the 'method' [100 3 6 6])
```

2.1.3 Other arguments

Other arguments such as minApx, realLength, keepOrder, showFig and writeT are easy to use. The details can be found in the main example above.

2.2 OUTPUT

In MATLAB, GiNA produces a structure with multiple measurments per fruit. Each parameter is contained in a MATLAB field structure. These are the measurments:

1. LvsW is the length / width of each object (including references).
2. blobLength is the length of each object (including references).
3. blobWidth is the width of each object (including references).
4. projectedArea is the two-dimensional area of each object (incuding references).
5. projectedPerimeter of each object (including references).
6. skinSurface is an estimation of the surface of each object (including references). This estimation was made considering that the object is an sphere or a egg. For irregular objects it does not work.
7. blobVolumen is an estimation of the volumen of each object (including references). This estimation was made considering that the object is an sphere or a egg. For irregular objects it does not work.
8. blobEccentricty is the eccentricty of each object (including references). Values closer to 1 are a indication of ciruclar objects.

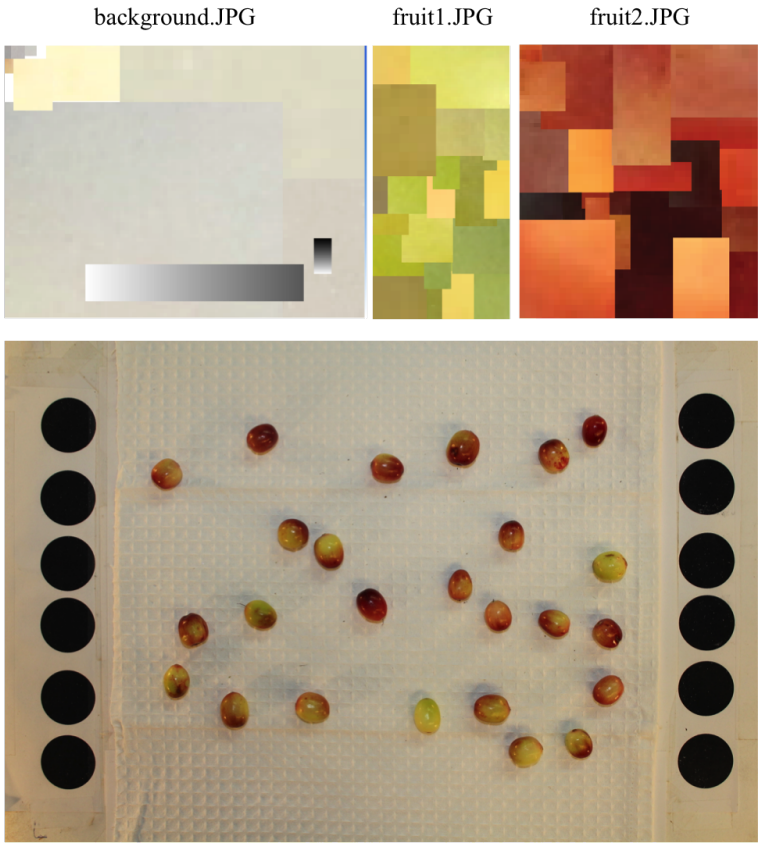


Figure 5: Neural netwrok approach for background extraction. These are examples of the pictures used as arguments in the function `netGiNA`

9. blobSolidity is the solidity of each object (including % references). Values closer to 1 are a indication of ciruclar objects.

10. RGBcolor contains a vector with 6 measurments related with color of each object (including references). The first three elements are the median values for the color channels R, G and B. The last three elements are the variance in the the color channels R, G and B.

11. bwColor contains the color (in grey scale) of each object (including references). vbwColor contains the variation in color (in grey scale) of each object (including references).

12. accuracy produces the length of each of the references used in the picture. It is a good way to see how good were the estimated measurments and object recognition. When the argument 'realLength' is used (see detailes below), the measurments are in the specific unit, when it is not used, the measurments is in pixels.

13. file indicates the name of the processed file.

14. After these measurments, there is another set of data fields with the same measurments adjusted by the references (measurments for the references are removedon these fields). The names for the data filed are the same except because a '_r' was added. Finally, when the argument 'realLength' is used (see details above), an additional set of data fields is generated ('_r2'). These measurments contain adjusted measurments using a specifc unit.