Getting Started with D.I.C.e

YOUDOM KEMMOE Victor

Contents

Prologue

Welcome

Welcome to *Get started with D.I.C.e*, an introductory guide written to help new users get started with D.I.C.e

Our goal is to cover the basic of D.I.C.e (Such as installation and working with for simple utilisation or for development). We designed this guide to be simple to follow, with step-by-step instructions and plenty of screenshots, allowing you to discover the potential of D.I.C.e.

Getting Started with D.I.C.e is not intended to be a comprehensive Digital image processing instruction manual. It is a quick-start guide that will get you doing the things you need to do with your computer easily, without getting bogged down with technical details. With the help of this guide, it should not take long before new users get used to the D.I.C.e desktop environment.

Description and Purposes

D.I.C.e or Digitam Image Code Experience is a free open source software developed to support students in their different courses implying Digital image processing or the treament of digital images; because performing manually some operations of digital image treatment can be quite difficult and confusing we thougth about given the students a tool to do those operations faster and simpler. It is made of 02 modules:

- A module for the treatment of real or concrete image: in this part of the program, users will be able to apply the different algortihms of digital image processing and see the effect on real image
- A module for the experimentation of different digital image processing image algortihm on matrixs of grayscales: here the user will be able to by himself creates a matrixs of grayscales representing an image and appreciates the different results of image processing algorithm.

Hence the purposes of D.I.C.e are:

- facilitates the decomposition in bands of an image
- shows the different method of displaying an image
- facilitatates the application of different filters on an image
- facilitates the application of some morphologycal operations
- facilitates the texture analysis of an image
- allows students to directly perform algorthims of image processing

1 Installation

Getting D.I.C.e

Before you can get started with D.I.C.e , you will need to obtain a copy of D.I.C.e (Jar file or Source code). Some option for doing this are outlined bellow.

Java Development Kit (JDK) is a software bundle that will help you to run a java application. For more information about JDK, visit:
http://www.oracle.com/technetwork/java/index.html

Roboto is a neo-grotesque sans-serif typeface family developed by Google as the system font for its mobile operating system Android. For more information, visit:
https://fonts.google.com/specimen/Roboto

An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development. For more information about intellij idea , visit:
https://www.jetbrains.com/idea/

Minimum system requirements

Below is a list of hardware and software specifications that your computer should meet as a minimum requirement.

Software:
- Java Development Kit 8.90 or ulterior version
- Windows OS or Mac OS or Linux OS distribution
- Roboto fonts

Hardware:
-200 MB of disk space
-2 GB of system memory (RAM)
-1.5 Ghz Core Duo processor

For developer:
- Intellij IDEA IDE (Recommended), or any IDE supporing java development
- JavaFx Scene Builder
- Java library: Sphinx4 core version 5prealpha, JFoenix

Downloading D.I.C.e

Your can get a copy of the Jar file at : //Link to be paste
For developer, you can get your copy of the source code by sending a mail at this address : youdom.k.v@gmail.com.

Installing D.I.C.e

For developers; at most 200MB will be needed (not including requirements). After getting the source code of the project and having installed the different requirement, you will have for those who installed intellij IDEA Open your IDE; click on open, look for the directory where you saved the source code; enter an open it. For those who installed another IDE than Intellij IDEA, you will have to create a new project in your IDE, copy the src directory from where you save the project and replace it with the src directory inside your project create with your IDE and include the different libraries included in the folder Library. For more you can easily find tutorial on how to open Intellij IDEA project on Eclipse or Netbeans. After that all you will have to do is compile the project and run it.

For non-developer, At most 200MB will be needed (not including requirements). After getting the jar file, go in the directory where the you saved it and just double click on it. It may appears that the jar file will not launch (still try to fix that issue) in that case, you can just follow the procedure for developer; what you will see may afraid you if you have don't have a little background in programming but don't worry; just each time launch the main method inside the Main class in the directory Main each time you will want to access the application. Don't modify something in what will appear; just run the main method. We will said it twice it is better to use Intellij IDEA to open the project in that case so that you will not have too much problem on how to transfert a project (application) from one IDE to Another.

2 D.I.C.e's Working Environment

In this section we are going to explore the working environment of D.I.C.e.

Home Page

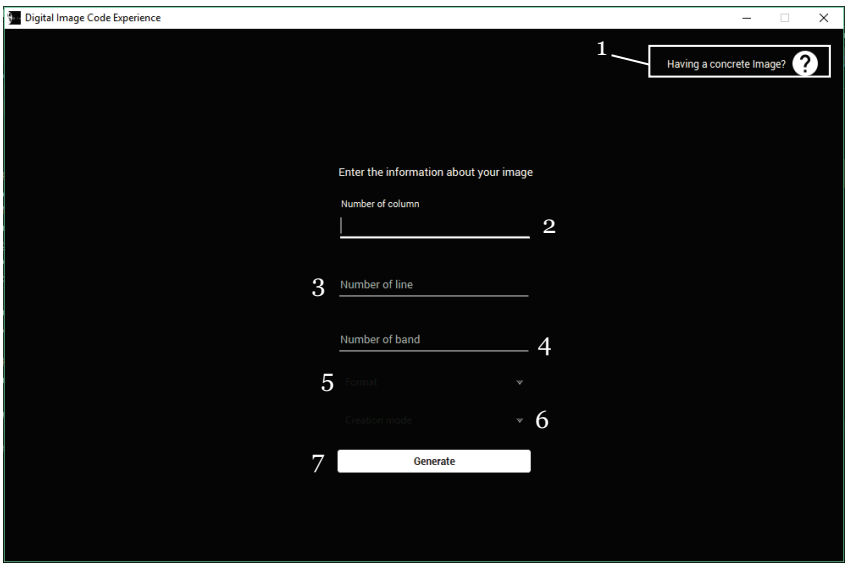


Fig 2.1: Home Page

1 : Button "Having a Concrete Image ?" It's used to access the second part of the program, the treatment of concrete Image
2: Text field where the user enters the number of columns of its matrix of grayscales
3: Text field where the user enters the number of lines of its matrix of grayscales
4: Textfield where the user enters the number of bands of its image
5: Selector where the user selects the formats of its image
6: Selector where the user selects the generating mode of its image, different modes are:
- Automatically : here the program generates random values of grayscales and assign it to the matrixs representing the image.
- Semi-automatically: here the program pop up a panel where the user will be able to enter by himself values of grayscales and if tired let the program fills the remaining values and/or modifies some.
- Manually: here the program pop up a panel where the user by himself fills all the values of grayscales.

7: Button "Generate", by clicking on that the program will collect all the necessary data and produces an "image" made of matrixs of grayscales on which user will be able to apply different image processing algortihm.

Module : Matrixs of grayscales

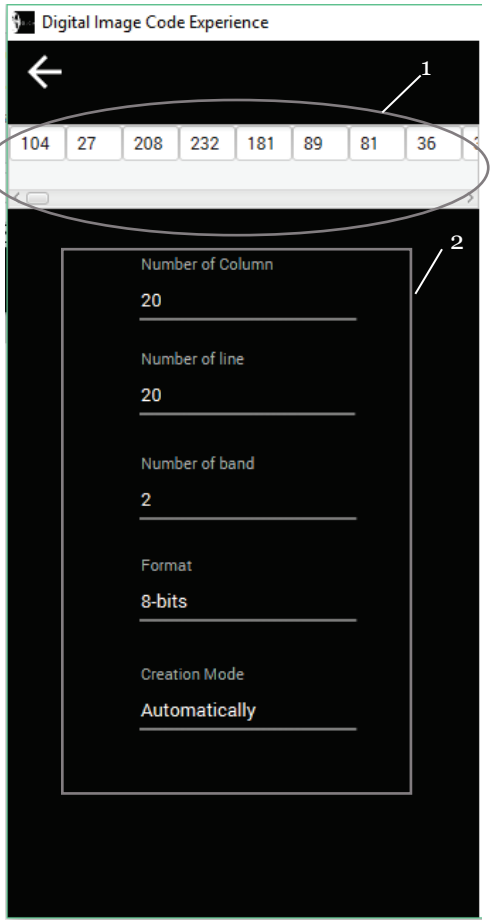


Fig 2.2: Info Tab

here is the info tab; in mode matrixs of grayscales, the program provides the user with an “info tab” where the user will always have different information about the created “image”.

1: represent the magnetic band; here the program stores the original magnetic band of the image so that the user can access it at anytime.

2: set of textfields keeping information about the orignal image such as number of colums, number of lines, number of bands ... of the original image

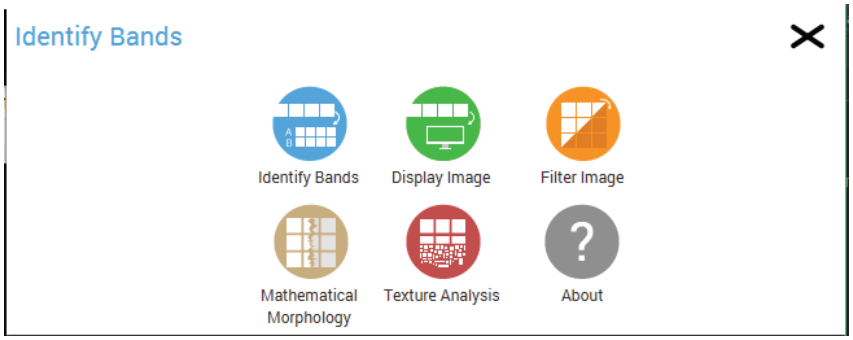


Fig 2.3: Menu

here is the menu, the user can navigate through the application and access different functionalities of the program through this menu.

- Identify bands: here the user can see the different bands compo-

sing its “image”.
- Display Image: here the user can perform some display algorithm of images such as pixel by pixel.
- Filter Image: here the user can perform some filtering algorithm on its image such as lee filter, sobel horizontal.
- Mathematical Morphology: here the user can perform some mathematical morphologycal operation such as erosion, dilation
- Texture analysis: here the user can perform some texture analysis such as Entropy on its image.

Module : Concrete Image

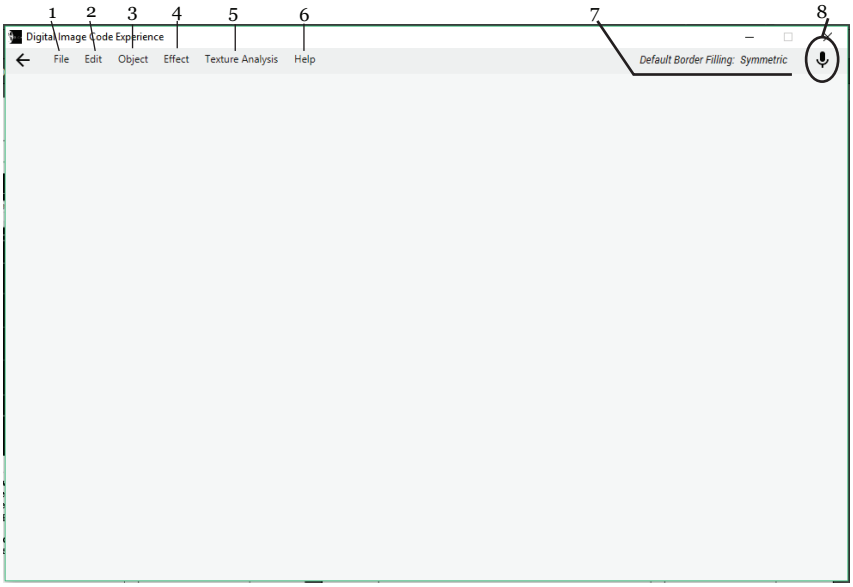


Fig 2.4: Concrete image module interface

1: File, this menu allows the user to open an image, save the treated image, close the application
2: Edit, this menu allows the user to undo a performed action or changed the default border filling method
3: Object, this menu allows the user to get information about the opened image
4: Effect, this menu allows the user to apply some effect on an image such as convertion to grayscale (black and white) or apply a filter on the image
5: Texture analysis, this menu allows the user to perform some texture analysis on an image.
6: Help, in this menu; user can get information about the application (version, author...)
7: represent the current default border filling selected; this field is used by vocal recognition to select the border filling method to user.
8: button to active the vocal recognition, some command can be perform through vocal command; this can be a plus for teachers who can use this option to perform remotely some commands instead of going from its computer to the board in other to explain something.

3 How To Use D.I.C.e

Module: Matrix of Grayscales.

Create an Image

To create an image made of matrixs of grayscales; the user just has to fill the different textfields and selectors of the homepage (should not click on the button “Having a concrete image?” this is to access the module handling true images) and click on generate.

Seperate Bands

This is done automatically by D.I.C.e after clicking on the button generate on the homepage. But if the user wants to resee the different bands he can just click on the button “Identify bands” from the menu

Display Image

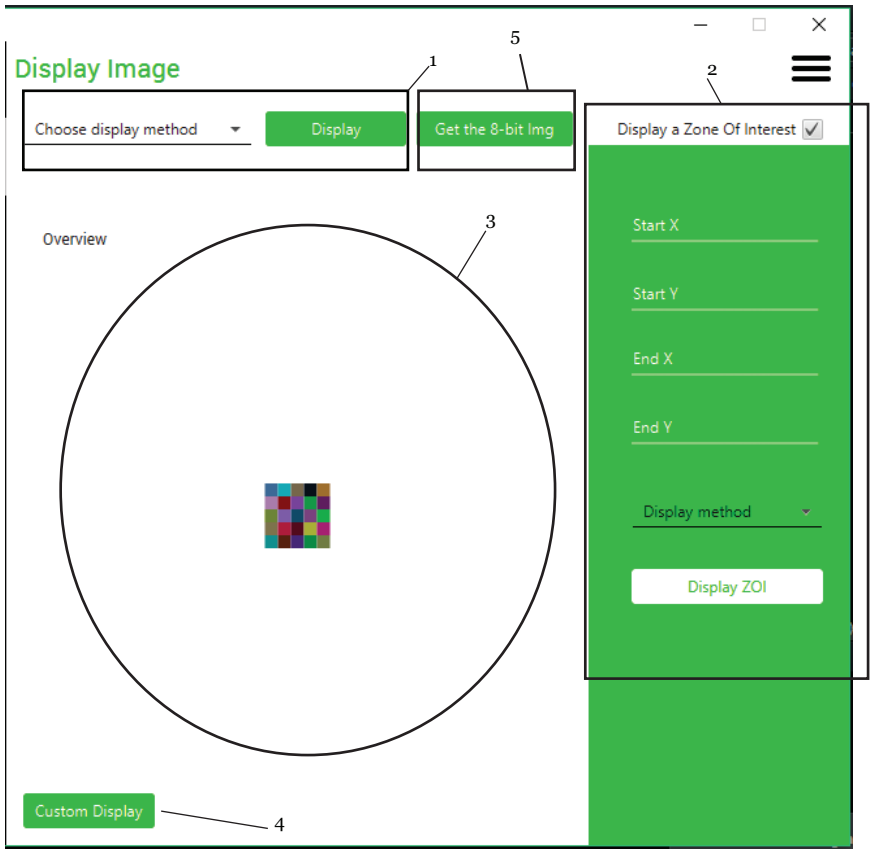


Fig 3.1: Display Image Interface

1: in this area the user select the display mode on the selector and click on Display to get an apercu of the different display’s algorithm.
2: in this area the user enters the different bounding values of its zone of interest and choose the displaying algorthim in other to obtain its zone of interest
3: in this area an overview of the image is presented

4: by clicking on “Custom display”; the user has the option of choosing the color that should be assign for a set grayscales; this option is only value for image of one band
5: here the user can get the 8-bit version of a 16-bit image; only available for a 16-bit image.

Filter an Image

A Filter can be viewed as a selector; it can be used to emphasize or remove certain characteristics of an image. Here the user is experiment linear and non-linear filter

Linear Filter

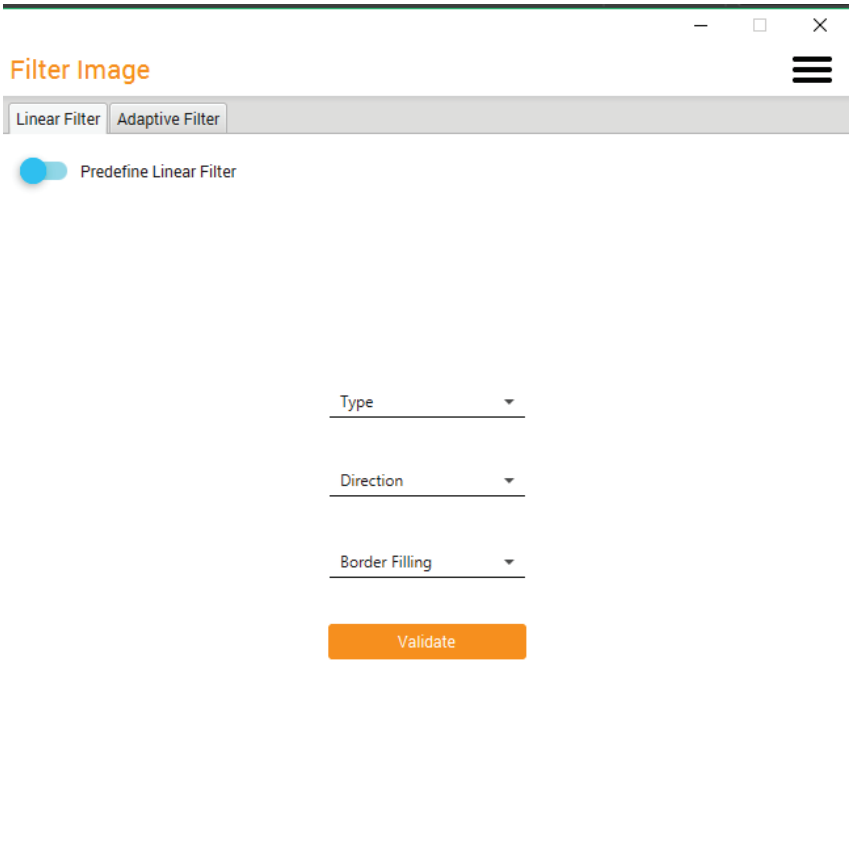


Fig 3.2: Linear Filter interface

D.I.C.e contains some already predefine filter that the user can just choose in selectors and click on validate.

Custom Linear Filter

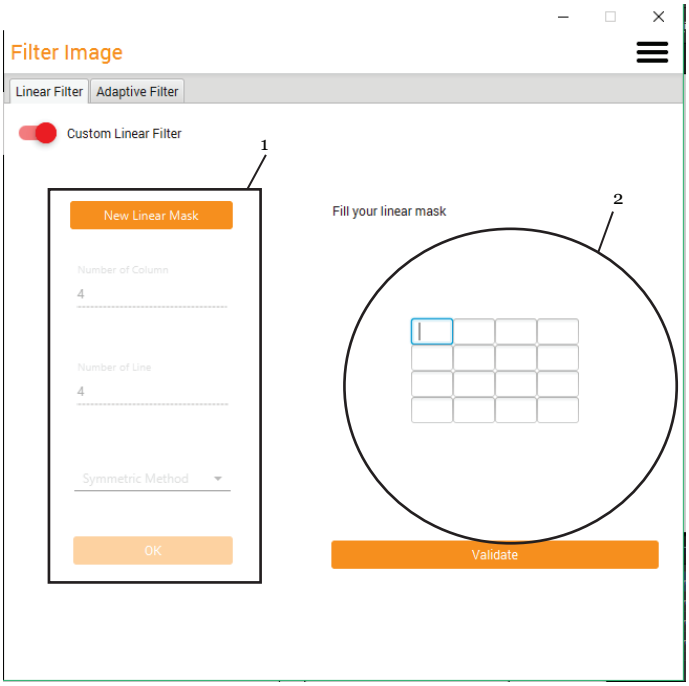


Fig 3.3: Custom Linear Filter Interface

Activated by switching to red the Toggle Button; the user can access an interface where he can defined its own linear filter an apply it on its image.

1: Where the user define the size of the filter and the border filling to be used

2: where the user define the different value of the the matrix of the filter.

Adaptive Filter

Adaptive filter in contrary to linear filter don't use matrixs of values to filter the image but statistics that are calculated.

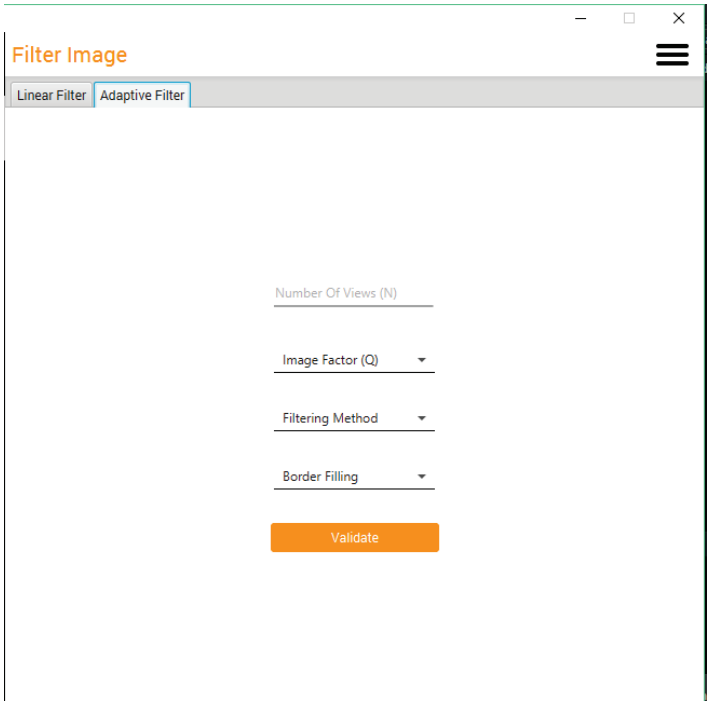


Fig 3.4: Adaptive Filter interface

D.I.C.e already comes with some pre-built adaptive filter.
The user can use any of them by entering the necessary values in the textfields and selectors.

Mathematical Morphology operation

The mathematical morphology is a technique of image analysis based on the notion of sets. While in Filtering we are using matrixs of values in mathematical Morphology operation we are using matrixs of filled with -1 to disable pixel and 1 to enable pixel, which in the end serves to represent geometrical forms.
The D.I.C.e comes with some pre-build morphologycal operations.

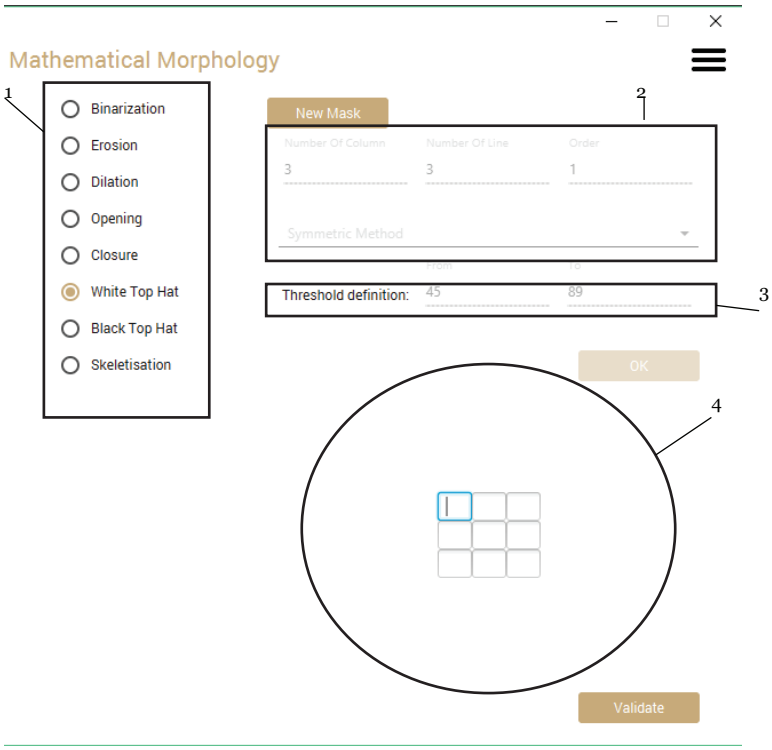


Fig 3.5: Mathematical Morphology interface

1: Set of available Mathematical Morpholgy operation; the user choose the one he wants to perform by selecting it.
2: Fields defining the size of the mask (matrixs of -1 and 1) and the border filling to be used
3: Fields defining threshold for binarization.
4: Area where the user fills the different value of its matrix in order to represent its mask.

Texture Analysis

Texture analysis is a method that takes into account the spatial distribution of greyscales in a given pixel neighbourhood in the image.

First Order Texture Analysis

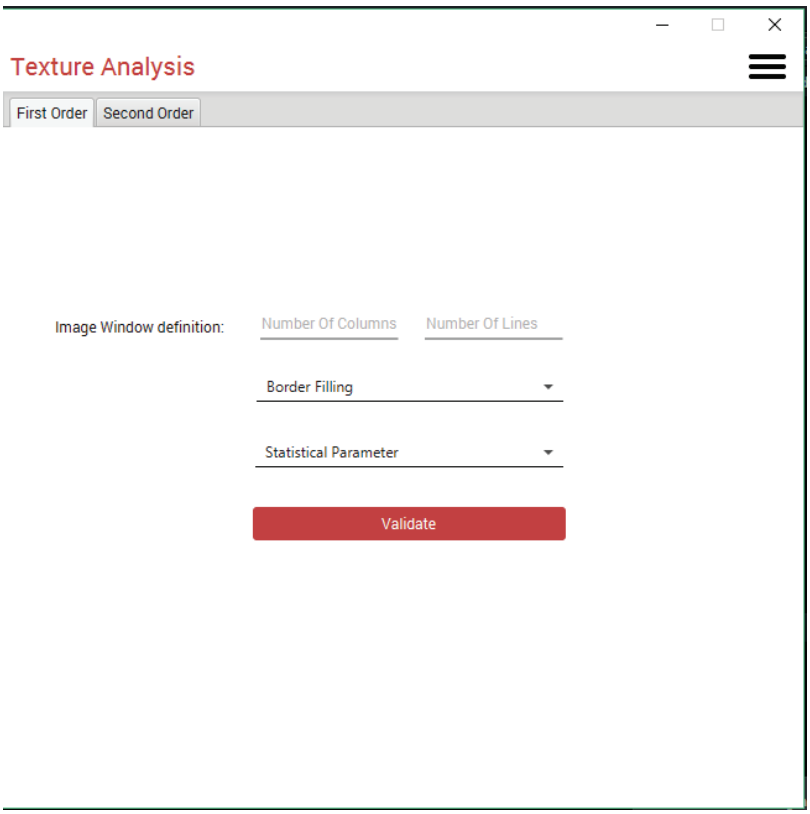


Fig 3.6: First Order Interface

First order statistical methods are based on first order histograms. Such a histogram specifies the frequency of appearance of a given greyscale in a given neighbourhood of the image called image window. D.I.C.e comes some pre-build first order texture analysis that the user will be able to use.
The user enters the values in the required fields; available statistical parameters are: standard deviation, energy, contrast, entropy.

Second Order Texture Analysis

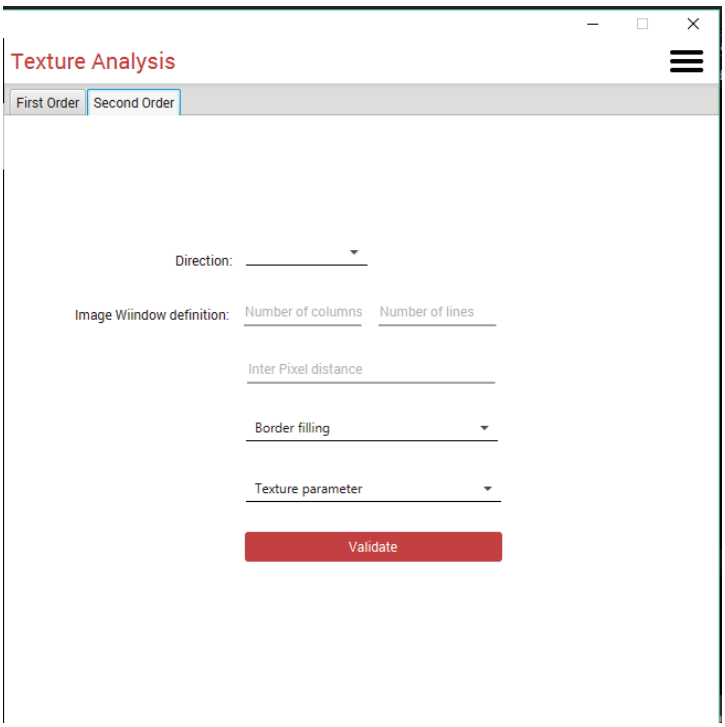


Fig 3.6: Second Order Interface

In contrary to First order texturing methods; these methods are based on the second order histograms called co-occurrence matrices. The co-occurrence matrix expresses the probability of appearance of each couple of greyscales (i,j) in the image window, with respect to a given direction and inter-pixels distance.
The user enters required parameters and validate. Available second parameters are Dissimilarity, Contrast, Entropy, Energy.

4 Learning More

What else can I do with D.I.C.e?

Well at this point you should now be able to use D.I.C.e to support you in your different courses talking about digital image processing. Also you can use the module working with a concrete image as a daily photo editor.

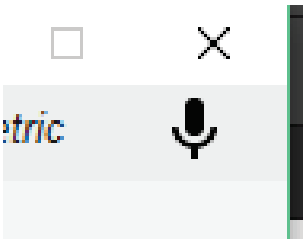
What is digital image processing?

Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing.

Open Source Software

Ubuntu is open source software (OSS). OSS differs from proprietary software. Proprietary software is defined as software whose source code is not freely available for modification or distribution by anyone but the rightsholder. Microsoft Windows and Adobe Photoshop are examples of proprietary software.

Some Voice Commands



Present in the module “Work on a concrete Image”; D.I.C.e provides some vocal commands through which users can perform some operations. As said earlier it can be very helpful for teachers or “lazy peoples”.
Those commands are:

convert to grayscale / dice convert to grayscale
convert to black and white/ dice convert to black and white:
to transform the image into a black and white image

convert to negative/ dice convert to negative: to apply the negative effect on the image

undo/ go back/ go back to the previous version: to go back to the previous process

perfrom es tization : to perform the skeletisation of the image

stop: to exit the voice recognition mode,

close/ close the application: Close the program

Glossary

IDE: integrated development environment.

Image window: rectangular or square portion of the image.

Mask: structuring element which is a simple geometrical object with a classical form such as: rectangle, circle, square, hexagon, octagon...