



## Mezzofanti

Augmenting reality through text-recognition.

<http://www.itwizard.ro> – more docs  
<http://code.google.com/p/mezzofanti/> - all code,  
Apache License 2.0

contact: [office@itwizard.ro](mailto:office@itwizard.ro)

## User Manual

Project Name	Mezzofanti
Project ID	SD_MEZ
Document ID	SD_MEZ_UM
Version	1.1
Date	
Status	Release
Classification	General Business Information

### ***IT Wizard General Business Information***

*The information contained here is considered General Business Information, and may be used as is without prior consent from IT Wizard. Any modifications of this text, or parts of it, will be made only with IT Wizard's written consent.*

Acceptance \_\_\_\_\_

# Version Tracking

Date	Author(s)	Revision	Comments
August 20 <sup>h</sup> , 2009	Mihai Mircea	1.1	Added Mezzofanti User Manual
July 19 <sup>th</sup> , 2005	Mihai Mircea	1.0	New template created

# Preface

This document is the *User Manual* of the Mezzofanti application.

- Scope** The *UM* specifies the system internal architecture, achieved features and interaction model.
- Rationale** This document is the result of the project finalization.
- Applicability** This document will be used for integration purposes, understanding the engine and using the software.
- Author** The members of the Project Team assigned by the Project Manager to write to this document will update and maintain this document.
- Readers** This document is useful to you if you are a stakeholder or a member of the Project Team.

# Table of Contents

Version Tracking .....	2
<b>1. Introduction .....</b>	<b>5</b>
1.1 Purpose .....	5
1.2 Description .....	5
1.2.1 Features .....	5
<b>2. The user interface .....</b>	<b>6</b>
2.1 Capture mode .....	6
2.2 Results mode .....	8
2.3 Preferences .....	11
<b>3. System architecture .....</b>	<b>12</b>
3.1 Basic tips for good text-recognition .....	13

# 1. Introduction

## 1.1 Purpose

This User Manual describes the Mezzofanti application.  
This document is organized into the following chapters.

Chapter	Contains
Chapter 1 – <b>Introduction</b>	A brief overview of the product, features and performances, and team-member responsibilities.
Chapter 2 – <b>The User Interface</b>	Overall system architecture, threading architecture, engine algorithms and a brief intro about the GPS architecture and signal properties.
Chapter 3 – <b>System modules</b>	The system modules implementation and features. Here we tackle all the design considerations for: acquisition, tracking, decoding and the computation modules.
Annexes	

## 1.2 Description

This is a brief document on how to use Mezzofanti software.

Mezzofanti is an Android based application, that augments reality through text-recognition.

The application at a glance has two modes of operation:

1. Select a text of interest (ex: menu at a restaurant, tourist information sign, etc), and take a photo. The internal-OCR engine will start Text Recognition. For the rest of the document this will be called **Capture-Mode**.
2. Use Mezzofanti to translate the text (or parts of it) to any one of over forty languages, or to google it on the web or on Wikipedia. For the rest of the document this will be called **Results-Mode**.

### 1.2.1 Features

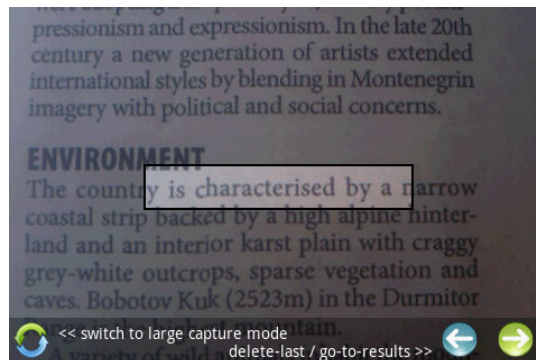
- Internal text-recognition engine, based on Tesseract 2.03 – developed by Google under Apache License 2.0.
- Line-mode / full-capture modes of operation, depending on the user's needs
- Translate in more than 40 languages, module based on Google Translate
- Email / SMS / google the text / wiki the text

## 2. The user Interface

### 2.1 Capture mode

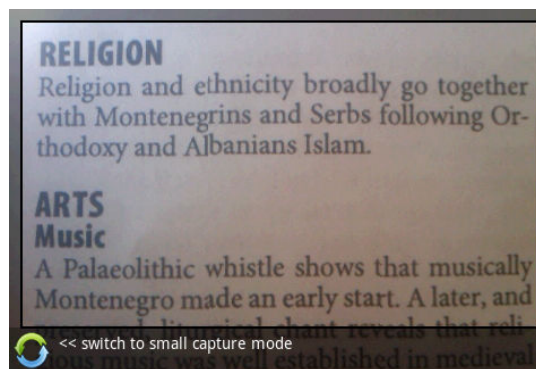
Within the capture-mode has two modes of operation line-mode and full-capture, to switch between them, just press the “switch-button” at the bottom-left of the screen.

In **line-mode** the user may text-recognize only a single line. The screen-shot below depicts this mode. The advantage is that this mode is very quick (seconds). So one should use this if one wants to input small amounts of data quickly.

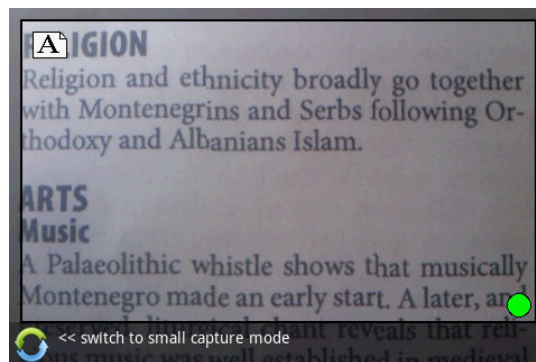


In **full-capture** mode, one can do text-recognition for the whole camera-view. The process may take between 5-30 seconds, depending especially on the amount of text to be recognized, and the light conditions.

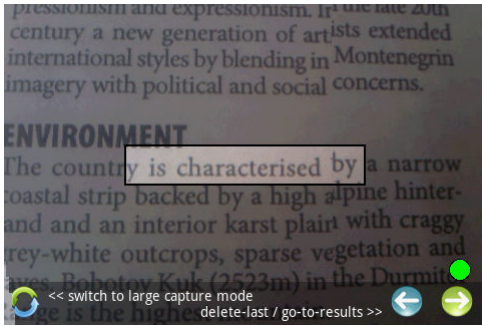
Note: the Mezzofanti OCR engine is based on a modified **Tesseract 2.03** version, developed currently by Google under Apache License 2.0. Tesseract is considered to be one of the very accurate OCR engines, but it's performance are influenced greatly by some simple tips (See tips of using section).



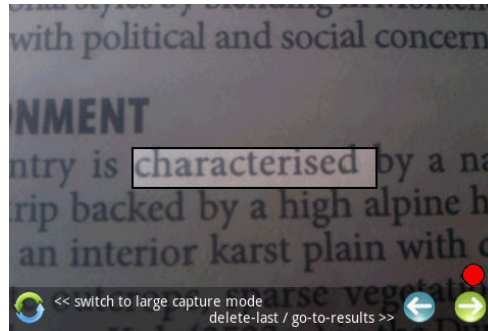
In full-mode, the user is also informed about the portrait/landscape mode of operation (at the top of the view).



In both operating modes, the user may start to text-recognize by placing the text of interest in the middle of the black box, and pressing the camera button to first focus, and than to take the picture. If the **camera is focused**, a green circle will appear at the bottom right of your screen; if the camera is **not focused** a red circle will appear instead and the phone will vibrate.



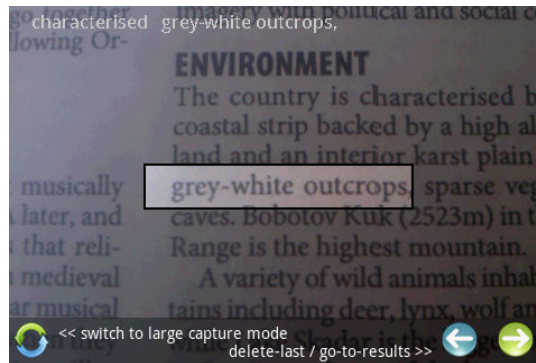
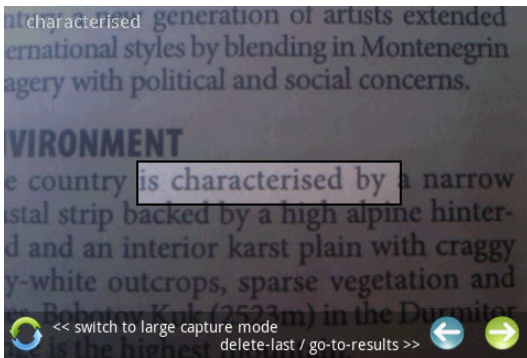
Focused camera (ok)



Unfocused camera (bad)

In **line-mode** the user can recognize text very fast, all text is added at the top of the screen (top picture).

The user may text-recognize several times, the new text will just add to the previous results, at the top of the screen.

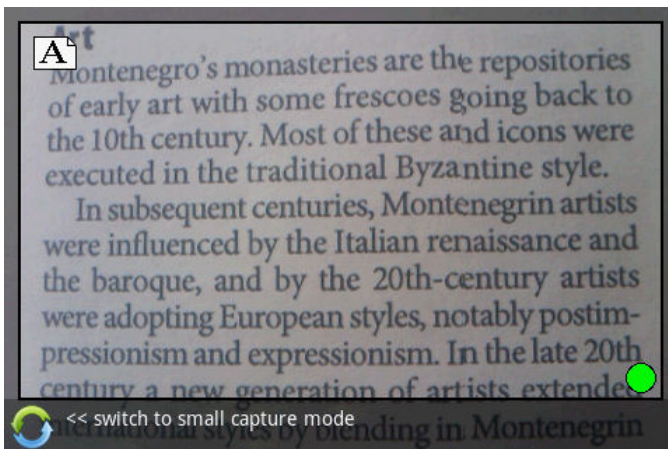


In line-mode, the user has two more buttons:

- delete-last : that deletes the last word in the results-list on top
- go-to-results : that will take all the results-lists and go to the results mode

Regardless you use the full-mode or the line-mode and then press 'go-to-results', the next stage is using your results, in results-mode.

Let's **text-recognize a larger text**, in order to exemplify the results-mode better.



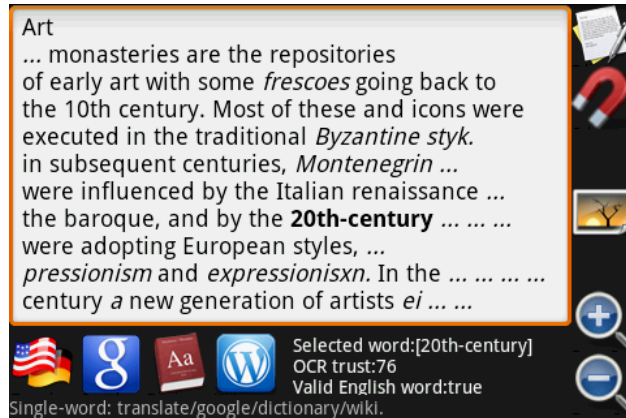
## 2.2 Results mode

Once prompted in the **results mode**, the user may see all recognized text.

The “...” **notation**, marks text that was not correctly recognized (this generally happens at the sides of the picture).

Pressing each word at a time will display the OCR trust for the word (ex: “20<sup>th</sup>-century” was recognized with a trust of 76%), and the validity of the word according to the internal dictionary.

**Marked in italics** are the invalid-dictionary words (either <1> the word is not in the dictionary – ex: “*frescoes*” and you may choose to add the word to the dictionary, or <2> the word was incorrectly recognized, ex: “Byzantine *styk*” instead of “style”).



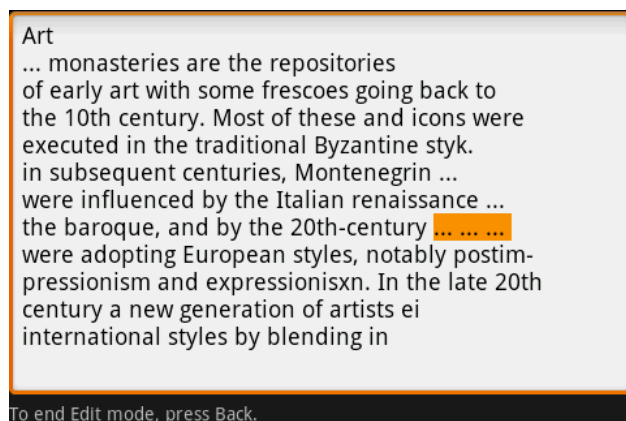
Results mode immediately after OCR, selecting a valid-dictionary composed word.



Selecting an invalid valid word.

In this case the user may consider modifying a bit the text, in the editor (top-right button).

This is the text editor, the user may select, add, delete text. Once the editing is done, the user will return to the results-mode, and will find that all italics annotations and the OCR internal information will be discarded.



Editing the text

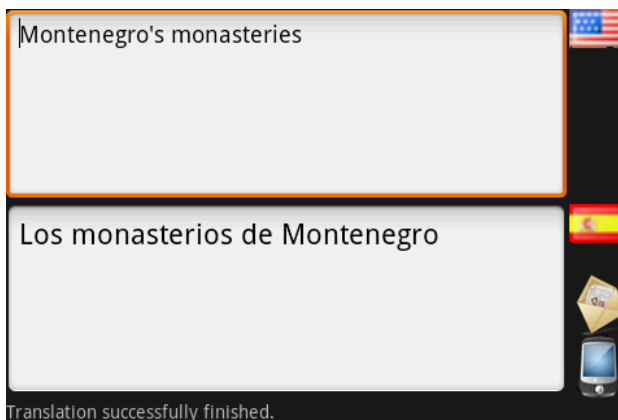
Just below the edit-button, is the **magnet button**. Using it the user may select more words. Once one or more words are selected, the user may:

- translate the phrase in one of the 40 available languages
- google the phrase, or search for images
- look in the dictionary
- search in wiki pages for more information (ex: montenegro's monasteries).



Back from editing, using the magnet

Now let's **translate** from English to Spanish. Once the translation is done, you can email or SMS your results.



Translation successfully finished.  
Translate English/Spanish

The language may be changed by simply pressing the flag. The translation will automatically restart once you finish the selection.



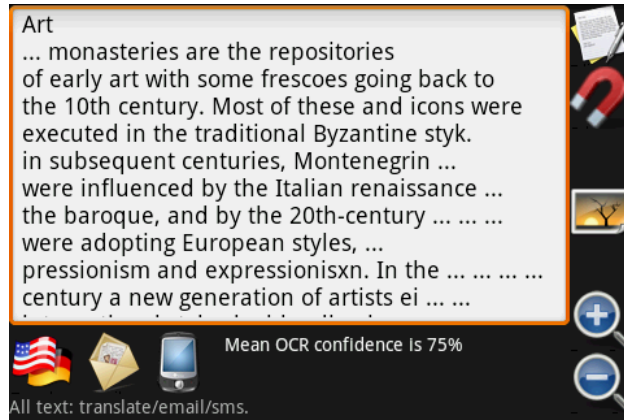
Translation successfully finished.  
Selecting the language

Returning to the main results, there are a few **other buttons** you may have noticed, if there is no selection made.

So if no text is selected, using the buttons at the bottom of the screen you can:

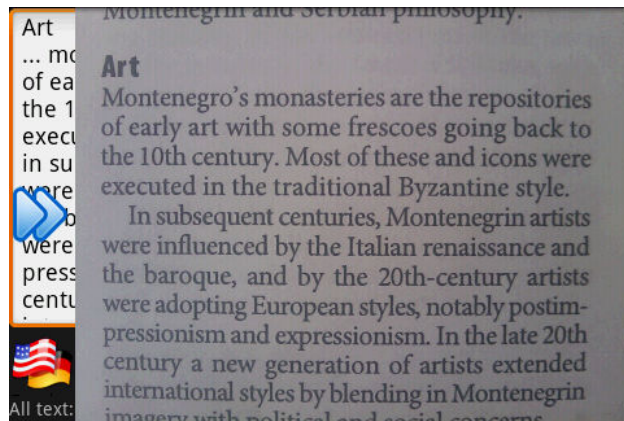
- **translate all text**
- **email / SMS** all text

Regardless of the selections made, at the right-bottom of the screen, there are the **zoom in-out** buttons.



Other buttons (no words are selected)

Also, maybe you noticed the **slider** at the right side of the screen, using it one may consult the original picture.



The sliding-drawer

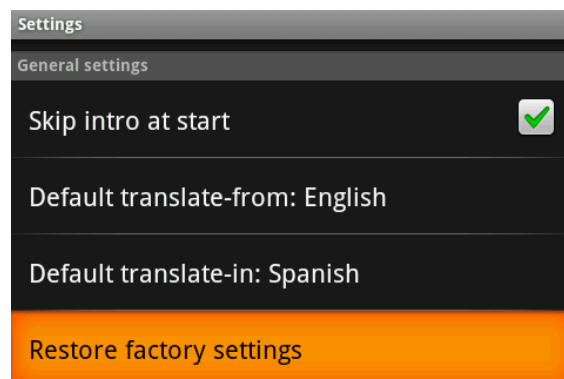
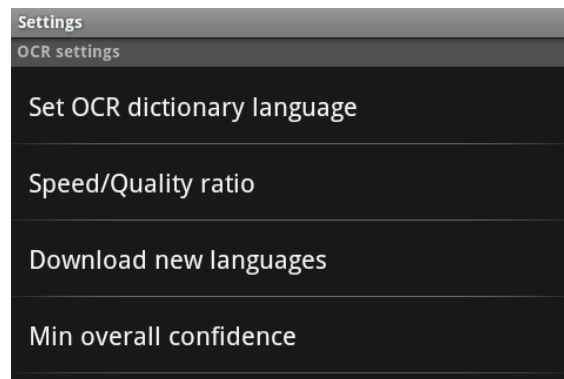
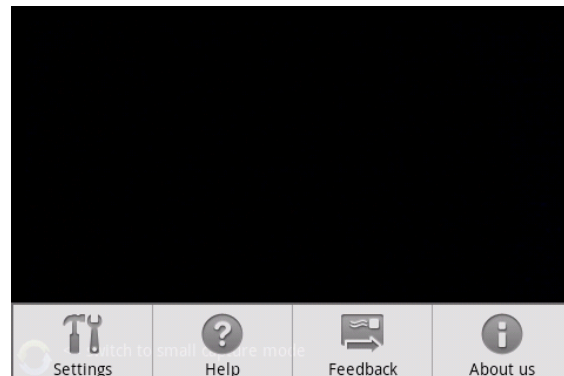
## 2.3 Preferences

Pressing 'Menu' button in the application will get the user in the Preferences. In the preferences the user may:

- change the applications settings
- ask for help
- send us his feedback
- get more information about the application and the company

The application settings will allow the user to:

- **“Set the OCR internal dictionary”** (currently supports 5 languages: English, French, Spanish, German and Italian)
- By default the application is installed only with English, if the user needs to add another language that he should press **“Download new languages”**
- the **“Speed / quality ratio”** will decrease the image resolution in order to get more speed
- the **“Minimal overall confidence”** is a barrier under which the whole recognized-text is considered invalid (a warning message will be displayed in results)
- the **“Minimal word confidence”** is a barrier under which a single word is considered improperly recognized (it will be shown in italics in the results)
- **“Use uneven light filter”** will enable/disable the specified filter. (if checked it is possible to add some 2-3 seconds to the processing time in full-capture mode)
- **“Skip intro at start”** will enable/disable the welcome screen at the start of the application
- **“Default translate-from / in”** are the default languages for the translation module
- **“Restore factory settings”** restores all parameters to the default values.

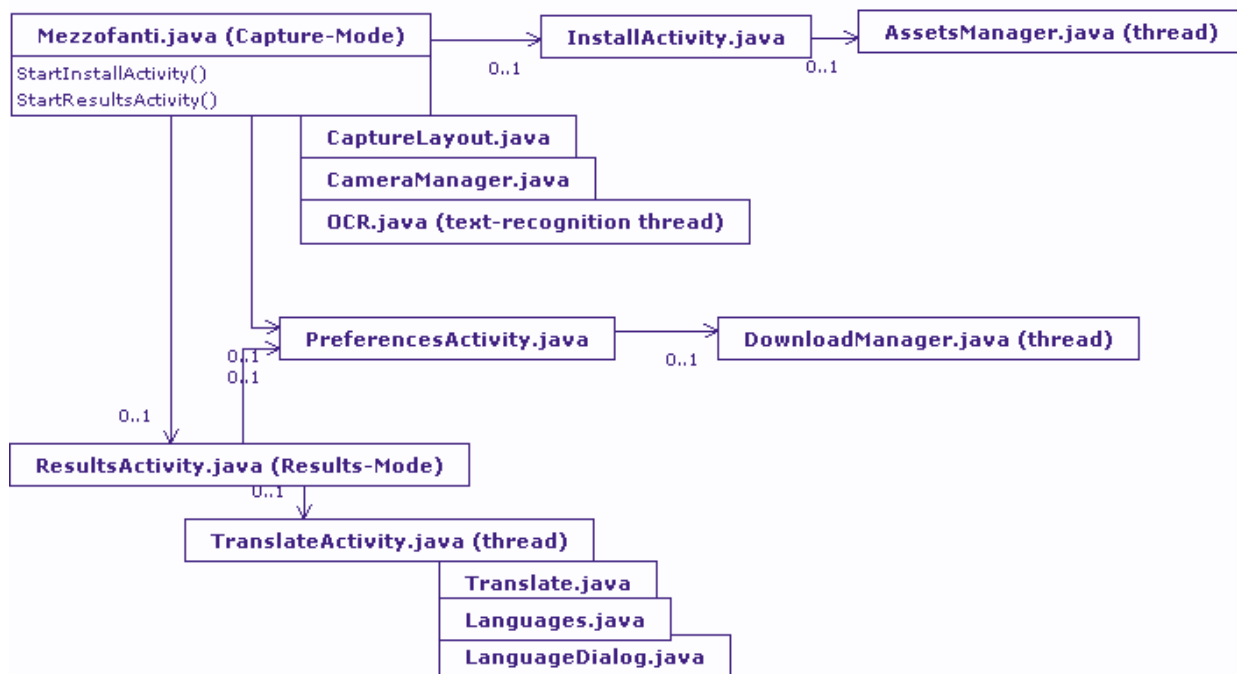


### 3. System architecture

The system is composed of 5 different activities:

- The Capture-Mode activity (Mezzofanti.java) - is responsible for the camera capture mode. It's auxiliary functions include the classes:
  - CaptureLayout.java – responsible for drawing the small/large capture boxes
  - CameraManager.java – is the Camera manager itself, with all the camera methods
  - OCR.java – the text-recognition wrapper
- ResultsActivity.java – responsible to present the results and start the auxiliary activities
- InstallActivity.java – displays the startup message with information about the application, also it installs the English language files. It's auxiliary functions include the classes:
  - AssetsManager.java – the assets installer (given the 1MB limitation of the assets class in the SDK, this class actually unzips the archive and copies the dictionary files)
- PreferencesActivity.java – started by both Mezzofanti.java and ResultsActivity.java, it is responsible for the settings-menu. It's auxiliary functions include the classes:
  - DownloadManager.java – downloads the language files from the internet repository.
- TranslateActivity.java – called from ResultsActivity.java, it is the wrapper for Google-translate. It's auxiliary functions include the classes: Translate.java, LanguageDialog.java, Languages.java, Maps.java

Thus the call graph looks like this:



Notes:

CustomImageButton.java is commonly used in all activities.

OnScreenHint.java is used to display an omnipresent message if important warnings are to be displayed (ex: sdcard missing)

### **3.1 Basic tips for good text-recognition**

1. Consider to take the picture in good lighting conditions (the OCR engines were not developed for camera use, but for ideal pictures taken in scanners). Phone cameras have poor image quality especially at night and using artificial light.
2. The simple fact that one can see the text, doesn't mean that the camera has a good shot of the picture, take care at the focus.
3. Try to have the sides of the text parallel to the camera, the text-recognition engine allows some degrees of freedom ( $< 10$  degrees) , consider not abusing them.