



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Scuola di Scienze Matematiche, Fisiche e Naturali  
Corso di Laurea in Informatica

Tesi di Laurea

LIBRERIA PYTHON PER L'OCR DI  
DOCUMENTI D'IDENTITÀ

PYTHON LIBRARY TO PERFORM OCR ON  
IDENTITY DOCUMENTS

ALESSIO FALAI

Relatore: *Maria Cecilia Verri*

Anno Accademico 2018-2019



---

## INDICE

---

<b>Introduzione</b>	<b>7</b>
<b>1 Prerequisiti matematici nell'ambito della computer vision</b>	<b>9</b>
1.1 Convoluzione	9
1.2 Operazioni morfologiche	9
<b>2 Binarizzazione di immagini</b>	<b>11</b>
2.1 Introduzione	11
2.2 Sogliatura globale	11
2.3 Sogliatura adattativa (o locale)	11
2.4 Sogliatura di Otsu	11
2.5 Algoritmo di sogliatura proposto	11
<b>3 Template matching</b>	<b>13</b>
3.1 Introduzione	13
3.2 SIFT	13
3.3 SURF	13
3.4 AKAZE	13
3.5 BRISK	13
<b>4 Text detection</b>	<b>15</b>
4.1 Introduzione	15
4.2 EAST	15
4.3 Algoritmo proposto	15
<b>5 Ulteriori migliorie</b>	<b>17</b>
5.1 Motori OCR	17
5.2 Benchmarks	17
5.3 Containerizzazione con Docker	17
5.4 Chiamate parametrizzabili sui campi del documento	17
5.5 Postprocessing	17
<b>Conclusioni e sviluppi futuri</b>	<b>19</b>
<b>Ringraziamenti</b>	<b>21</b>



---

## ELENCO DELLE FIGURE

---



*"If a machine is expected to be infallible, it cannot also be intelligent"*  
— Alan Turing





---

## INTRODUZIONE

---

Questa tesi vuole essere un riassunto delle attività svolte in un tirocinio curriculare presso l'azienda QI-LAB di Firenze nel periodo *marzo - giugno 2019*. L'offerta di tirocinio prevedeva l'apporto di migliorie, in termini di accuratezza ed efficienza, riguardanti una libreria per il riconoscimento ottico dei caratteri di alcuni campi di documenti d'identità, attualmente solo italiani. La libreria, denominata QI-OCR, era inizialmente il risultato del lavoro compiuto da un precedente tirocinante, *Emilio Cecchini*, che si era occupato dell'implementazione del framework di base, sul quale ho avuto il piacere di lavorare io stesso. QI-OCR viene distribuita come pacchetto *pip* ed è stata scritta con linguaggio Python, in versione 3.6.8, anche se compatibile con la versione minor successiva, ovvero la 3.7.x, nonché l'ultima versione stabilita rilasciata, nella data in cui mi trovo a scrivere questo testo. QI-OCR prevedeva la suddivisione in 4 moduli principali:

1. **Preprocessing:** Si occupa della localizzazione del documento all'interno dell'immagine, del ritaglio dei contorni e del raddrizzamento, tramite l'algoritmo di template matching *SIFT*.
2. **Docfields:** Si occupa di effettuare un ritaglio statico, mediante coordinate predeterminate, dei campi d'interesse del documento già centrato.
3. **OCR:** Si occupa di effettuare una binarizzazione dell'immagine mediante un algoritmo di sogliatura globale *ad-hoc* e di fornire l'immagine prodotta in input a un motore di OCR.
4. **Postprocessing:** Si occupa di applicare tecniche di analisi sintattica e semantica sui risultati ritornati dal software OCR.

Il lavoro da me svolto riguarda il miglioramento della libreria QI-OCR, mediante tecniche che verranno descritte successivamente. A tal proposito, la tesi è suddivisa in quattro capitoli. Il capitolo 2 riguarda lo studio e l'implementazione di tecniche di binarizzazione e segmentazione di immagini, ovvero soluzioni necessarie per la "pulizia" di una generica immagine contenente testo, che permettano di ottenere idealmente il solo

testo di colore nero e tutto il resto di colore bianco. Il capitolo 3 riguarda invece lo studio di algoritmi di segmentazione basata dal riscontro sul modello, in cui vengono introdotti metodi alternativi al ben noto SIFT per l'individuazione della posizione di un documento all'interno di un'immagine. Il capitolo 4 introduce l'applicazione di tecniche avanzate di *text-detection* con lo scopo di colmare problemi derivanti da documenti che non rispettano alcun tipo di template prefissato, come ad esempio la vecchia carta d'identità cartacea italiana, che risulta comunque rilasciabile, in alternativa alla nuova carta d'identità elettronica italiana, in casi di estrema esigenza o per tutti i cittadini che non possono recarsi per motivi di handicap presso il municipio. Infine, il capitolo 5 descrive ulteriori migliorie e innovazioni introdotte, sia per quanto riguarda il lato sviluppo/distribuzione (containerizzazione, benchmarks, ...), sia per quanto riguarda l'esperienza utente (OCR parametrizzabile sui campi del documento, implementazione di diversi motori di OCR, ...).

---

## PREREQUISITI MATEMATICI NELL'AMBITO DELLA COMPUTER VISION

---

### 1.1 CONVOLUZIONE

### 1.2 OPERAZIONI MORFOLOGICHE



---

## BINARIZZAZIONE DI IMMAGINI

---

### 2.1 INTRODUZIONE

### 2.2 SOGLIATURA GLOBALE

### 2.3 SOGLIATURA ADATTATIVA (O LOCALE)

### 2.4 SOGLIATURA DI OTSU

### 2.5 ALGORITMO DI SOGLIATURA PROPOSTO



# 3

---

## TEMPLATE MATCHING

---

### 3.1 INTRODUZIONE

### 3.2 SIFT

### 3.3 SURF

### 3.4 AKAZE

### 3.5 BRISK





---

## TEXT DETECTION

---

### 4.1 INTRODUZIONE

### 4.2 EAST

### 4.3 ALGORITMO PROPOSTO



# 5

---

## ULTERIORI MIGLIORIE

---

5.1 MOTORI OCR

5.2 BENCHMARKS

5.3 CONTAINERIZZAZIONE CON DOCKER

5.4 CHIAMATE PARAMETRIZZABILI SUI CAMPI DEL DOCUMENTO

5.5 POSTPROCESSING



---

## CONCLUSIONI E SVILUPPI FUTURI

---



---

## RINGRAZIAMENTI

---





---

## BIBLIOGRAFIA

---

[1] Autore - *titolo*

[2] Autore - *Titolo* - altre informazioni