

IMAGine

Francesco Paolo Castiglione, Davide Iraci, Andrea Montemaggiore

September 2020

Contents

1	Introduzione	2
2	Introduzione	3

1 Introduzione

IMAGine (IMAge-enGINE) è un linguaggio ideato e pensato per l'elaborazione delle immagini a trecentosessantasei gradi. Il linguaggio consente le elaborazioni più comuni attraverso un processo immediato ed intuitivo. E' stato pensato e sviluppato tenendo conto di alcuni fattori chiave quali semplicità d'uso e rapidità di apprendimento, basandosi anche su altri linguaggi già esistenti quali MATLAB, Python o software per la modifica di immagini.

Il resto della relazione è strutturato come segue: la sezione 2 illustra il contesto in cui si colloca il linguaggio proposto, nella sezione 3 vengono analizzati i dettagli implementativi, nella sezione 4 si illustrano le caratteristiche di IMAGine e degli esempi di applicazioni, infine nella sezione 5 vengono riepilogati i punti principali del lavoro svolto.

2 Stato dell'arte

Attualmente l'Image Processing avviene principalmente per mezzo di software già pronti quali Photoshop, Gimp, etc. Questi software sono stati pensati per un utilizzo da parte di amanti delle immagini e per professionisti in campo fotografico. Alcuni di questi, come il già citato Adobe Photoshop vengono proposti in una suite software chiamata "Adobe Creative 2020", la quale, nella sua ultima edizione, comprende altri applicativi per l'Image Processing di livello sempre più alto. Per lo sviluppo del progetto ci siamo principalmente basati sullo stato dell'arte dei linguaggi che permettono operazioni su immagini e sulle operazioni che permettono: da semplici rotazioni, zooming e shrinking alle più complicate operazioni di edge detection, cambio dello spazio di colori, etc; Per lo sviluppo di una sintassi concisa ed efficace sono stati individuati principalmente C, C++, Java e Python, tuttavia ognuno di essi ha pro e contro ma principalmente tutti richiedono l'utilizzo di librerie esterne per l'implementazione delle modifiche e operazioni su immagini. Per quanto riguarda C e Python sono attualmente i più veloci tuttavia C possiede un linguaggio complicato per l'utilizzo con ognuna delle librerie presenti per mentre Python possiede numerose librerie con una sintassi di facile utilizzo ma alcune di esse, o meglio quasi tutte, non ottengono un buon risultato in alcune operazioni più complicate come edge detection, tuttavia va sottolineato come "skimage" sia uno dei migliori in termini di prestazioni e risultati. Per C e C++ la libreria migliore in termini di prestazioni è "LibVips" adoperata anche in questo progetto, tuttavia possiede una sintassi di difficile utilizzo per via dei lunghi nomi dei metodi pre esistenti; L'ultima analisi si basa su Matlab, ampiamente usato, specialmente in ambito accademico, è il linguaggio per immagini per eccellenza tuttavia alcune operazioni, come media adattiva, mediano adattivo, etc, richiedono che il programmatore sappia a memoria il codice da scrivere, ovviamente operazioni più semplici come media e mediano classici sono implementati tramite funzioni built-in di Matlab; Dallo stato dell'arte vengono dunque individuati i passaggi chiave per la creazione del linguaggio: sintassi semplici e prestazioni d'alto livello.