

PCTO BRETON SPA

PETTENUZZO RICCARDO

L'azienda *Breton* necessita che il processo di controllo dell'usura del disco dentellato usato nella macchina per il taglio delle lastre, venga automatizzato grazie ad un programma sviluppato in *Python*. Il programma elabora la foto dello sfrido scattata da una fotocamera montata sulla macchina, viene calcolata la **differenza di spessore** lungo lo *sfrido* e quando quest'ultima oltrepassa un valore nell'ordine dei centesimi di millimetro, l'utensile necessita di essere sostituito.

1 Obiettivo e Descrizione del Progetto

L'azienda **Breton** dispone di macchine impiegate nel taglio di lastre di vario materiale, una di esse utilizza un *mandrino*⁰ che sostiene un disco dentellato. La lastra viene posizionata nel piano della macchina, sorretta da delle ventose che ne bloccano il movimento, quando la macchina viene azionata parte il programma di taglio inserito, il disco scende e grazie ad una soluzione acquosa la lastra viene tagliata. Il disco è soggetto ad **usura**, che viene periodicamente controllata da un'addetto esperto e la sostituzione dell'utensile avviene quando la larghezza dello *sfrido*¹ differisce nella lunghezza di quest'ultimo. Per permettere all'addetto di controllare lo stato dello strumento la lastra non viene tagliata completamente nella sua profondità.

L'obiettivo del progetto è quello di **automatizzare** il processo di controllo dello stato di usura del disco, quindi realizzare un programma in Python² che elabora una foto scattata da una fotocamera montata sulla macchina, in modo da ottenere la *differenza di spessore* lungo lo sfrido, il manutentore interviene nel caso in cui la differenza di spessore sia sopra un certo valore in centesimi di millimetro.

Il programma utilizza la libreria **OpenCV**³ per lo scatto della foto, successivamente per la sua elaborazione e infine per la visione finale del processo eseguiti per il ricavo del risultato.

⁰ Attrezzo della macchina operatrice con funzione di sostegno per l'utensile, o di supporto per il pezzo durante la lavorazione.

¹ Lo sfrido è lo scarto di lavorazione delle piastrelle che si crea durante la fase di posa.

² Python è un linguaggio di programmazione utilizzato nelle applicazioni web, sviluppo software, data science e machine learning

³ OpenCV è una potente libreria che consente di eseguire operazioni di elaborazione delle immagini, deep learning, machine learning e computer vision su feed video in diretta.

Il primo metodo di soluzione converte prima l'immagine in *Scala di Grigio* applica *Blur* e *Threshold* all'immagine, con l'obiettivo di ottenere solo la parte dello sfrido. Successivamente calcola lo spessore in base a come è orientato lo sfrido: se **verticale** calcola la differenza tra le coordinate in x di due punti alla stessa altezza. Se **obliquo**, trova prima l'equazione della *retta perpendicolare* allo sfrido e applica *pitagora* per trovare lo spessore dello sfrido. Se *orizzontale* ruota lo scatto e applica uno dei metodi di calcolo dello spessore precedenti. Infine itera il processo di calcolo in vari punti nell'altezza dell'immagine

2 Primo metodo di Soluzione del problema

Il primo metodo di soluzione del problema parte dal presupposto che l'immagine deve essere *filtrata* in modo da rendere visibile solamente la parte dello sfrido, questo processo avviene convertendo l'immagine scattata da **RGB** a **Scala di Grigi**⁴. Successivamente si utilizza prima la funzione *Blur* la quale sfuoca zone di pixel, quindi rende lo scatto più facile da filtrare grazie alla funzione *Threshold* che in base a un valore statico converte tutti i pixel che superano questo valore in bianco e tutti gli altri in nero, così da evidenziare nell'immagine solo lo sfrido.

Il passo successivo è quello di calcolare il numero di pixel della parte nera, ovvero lo sfrido, il programma partendo dalla metà dell'immagine cerca i *bordi* dello sfrido, in questo caso i bordi si trovano quando c'è la sequenza bianco e nero, oppure nero e bianco. Dopo di questo in base a come è **orientato** lo sfrido viene calcolato il numero di pixel tra i due bordi: nel caso in cui lo sfrido sia *verticale*, il programma calcola la differenza tra le coordinate in x dei due bordi. Nel caso in cui invece lo sfrido sia *obliquo*, il programma calcola prima l'**equazione** della retta adiacente allo sfrido, sfruttando i due punti nel bordo dello sfrido a diverse altezze dell'immagine, applica poi il **sistema di equazioni**⁵ passanti per due punti e trova l'equazione. Utilizza poi il **teorema delle rette perpendicolari**⁶, sapendo che rette perpendicolari tra loro presentano coefficienti angolari *reciproci*, trova poi l'equazione completa sostituendo le coordinate del punto nel bordo, successivamente cerca le coordinate del punto appartenente alla retta nel bordo opposto dello sfrido. Infine applica **pitagora** tra due punti per trovare la lunghezza del segmento. Nel caso in cui l'orientamento dello sfrido sia *orizzontale* il programma trova solo pixel bianchi e quindi ruota l'immagine di novanta gradi fino a quando non trova il bordo.

⁴ Ogni pixel dell'immagine viene convertito da una serie di tre valori a un valore unico da 0 a 255 in base all'intensità di quel pixel

⁵ In matematica, un sistema di equazioni è un insieme di due o più equazioni che ammettono le stesse soluzioni

⁶ Data una retta r ed un punto p, esiste ed è unica la retta che passa per p ed è perpendicolare ad r

In conclusione itera⁷ questo processo in diverse altezze dell'immagine e calcola poi lo *Spessore Medio*, la *Differenza di Spessore* tra lo spessore nel punto più alto e quello più basso e calcola poi la *Deviazione Standard*⁸ tra tutti gli spessori.

Il secondo metodo converte l'immagine in scala di grigi e non applica alcun filtro in quanto potrebbero compromettere l'integrità dell'immagine, vengono ricavati i **profili** della parte più alta dell'immagine e di quella più bassa, viene poi rilevato lo sfrido nel profilo e viene calcolato il punto medio dello sfrido, utilizza poi il sistema di equazioni e il teorema delle rette perpendicolari per ottenere l'equazione della retta perpendicolare allo sfrido.

Il programma applica vari *fattori di traslazione* per poi calcolarne il profilo. Nei profili trovati viene rilevata la parte dello sfrido e viene asportata, poi il profilo viene **approssimato ai bordi** e si calcola lo *spessore a metà altezza*, infine vengono calcolati spessore medio, differenza di spessore e deviazione standard

3 Secondo metodo di Soluzione del problema

Il secondo metodo di soluzione a differenza del primo non applica filtri all'immagine, in quanto potrebbero corrompere l'intergrità dei pixel dello sfrido, l'immagine viene solamente convertita in scala di grigi. Il procedimento di calcolo dello spessore dello sfrido parte prendendo tutti i valori dei pixel nella linea più alta dell'immagine e in quella più bassa. Grazie alla libreria **MatPlotLib**⁹ che permette di visionare grafici, si costruiscono due grafici che nell'asse delle ascisse i pixel e nell'asse delle ordinate il corrispettivo valore in scala di grigi, così da ottenere il **profilo** di quella serie di pixel. Sfruttando questi profili il programma capisce dove si trova lo sfrido e trova in entrambi i profili il *punto centrale*, come nel primo metodo si applica il *sistema di equazioni* e viene calcolata l'equazione dello sfrido, viene riapplicato il *teorema delle rette perpendicolari* e utilizzando un punto a metà altezza dell'immagine viene calcolata l'equazione della retta perpendicolare. Vengono poi utilizzati diversi **fattori di traslazione**¹⁰ così da avere le equazioni della stessa retta traslata in tutta l'altezza dell'immagine.

Dopo aver ottenuto le equazioni delle rette, utilizzo quest'ultime per ottenere i valori di ogni pixel di quella retta e si riutilizza *matplotlib* per costruire i grafici dei profili, per ogni profilo ottenuto viene rilevata la posizione dello sfrido e si ricostruisce il grafico del profilo solamente dello sfrido.

⁷ Ripetere. Un procedimento eseguito più volte

⁸ La deviazione standard è un indice riassuntivo delle differenze dei valori rispetto alla media della variabile.

⁹ Libreria per la creazione di grafici per il linguaggio di programmazione Python

¹⁰ È un valore positivo o negativo che trasla/sposta in altezza la retta principale

Infine per ottenere lo spessore dello sfrido non viene calcolato nè nel punto più alto del bordo nè nel punto più basso, si calcola dunque lo ***Spessore a Metà Altezza***¹¹, per fare questo quindi viene trovato il punto nei bordi in cui il valore del pixel cambia drasticamente questo punto viene poi approssimato al valore massimo, viene presa la metà di quel valore in entrambi i bordi e quindi si calcola lo *spessore a metà altezza*.

In conclusione questo processo viene ripetuto per ogni profilo trovato e vengono poi calcolati lo spessore medio, la differenza di spessore e infine la deviazione standard.

4 GUI del Progetto

La ***GUI***¹² del progetto è stata sviluppata utilizzando la libreria ***CustomTkinter***¹³, permette di scattare una foto, utilizzando una fotocamera esterna oppure quella integrata se presente, che poi viene salvata nel progetto e visionata nell'interfaccia. Quando si fa partire il programma questa foto viene utilizzata e l'interfaccia poi mostra i risultati numerici e l'immagine elaborata nei due diversi metodi di soluzione.

5 Conclusioni Finali

Il progetto sviluppato presenta due soluzioni, la prima dipende da come viene filtrata l'immagine, l'altra soluzione dipende principalmente dalla lastra. Entrambe le soluzioni raggiungono un risultato che non sempre è corretto: i filtri usati nel primo metodo possono non filtrare l'immagine completamente e quindi compromettere il metodo di calcolo, quando la lastra presenta cambi di intensità e colore il secondo metodo potrebbe non dare un risultato corretto. Alcune prove fatte in entrambi i metodi hanno dato buoni risultati e facendo un test con una fotocamera i risultati sono stati ottimi ed è stata calcolata la misura in millimetri di un pixel, circa due centesimi di millimetro.

Link Repository GitHub:

6 Esperienza PCTO

L'esperienza PCTO è stata molto bella e soprattutto utile, mi ha permesso di entrare un po' più nell'ottica di come funziona un'azienda, ho capito come ci si pone di fronte ad un problema reale e come un'azienda si interfaccia con aziende esterne. Grazie a questa esperienza ho arricchito le mie conoscenze e conosciuto persone fantastiche.

Ringrazio: Federico Milan, Luca Malorzo, De Zuani Stefano e l'intero DIGI Hub per l'esperienza!

¹¹ Viene calcolato lo spessore a metà altezza per ottenere un valore non troppo elevato per errore del bordo e non troppo basso per la differenza di spessore del fondo dello sfrido

¹² Graphical User Interface

¹³ CustomTkinter is a python desktop UI-library based on Tkinter