

UNIVERSITÀ  
degli STUDI  
di CATANIA

# UNIVERSITÀ DEGLI STUDI DI CATANIA

DIPARTIMENTO DI MATEMATICA E INFORMATICA

CORSO DI LAUREA TRIENNALE IN INFORMATICA

---

*Giulia Meo*

## **Monitoraggio e verifica delle procedure attraverso la comprendizione delle sequenze di interazioni tra uomo e oggetto**

---

RELAZIONE PROGETTO FINALE

---

Relatore: Prof. G. M. Farinella

Correlatore: Prof. F. Ragusa

Correlatore: Dott. R. Leonardi

---

Anno Accademico 2022 - 2023

# Contents

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Related Work</b>	<b>5</b>
2.1	Human-Object Interaction detection . . . . .	5
2.1.1	Algoritmi di Human-Object Interaction detection da Third Person point of View . . . . .	6
2.2	Algoritmi di Human-Object Interaction da First Person (Egocentric) View . . . . .	8
2.2.1	Egocentric Human-Object Interaction Detection Exploiting Synthetic Data . . . . .	12
2.3	Applicazioni esistenti a supporto degli operatori nel dominio industriale . . . . .	15
<b>3</b>	<b>Software Sviluppato</b>	<b>18</b>
3.1	Tecnologie utilizzate . . . . .	18
3.1.1	Libreria per la gestione delle interfacce grafiche . . . . .	18
3.1.2	Libreria per la gestione del supporto vocale . . . . .	20
3.1.3	Libreria per la gestione delle immagini . . . . .	20
3.2	Laboratorio ENIGMA . . . . .	22
3.3	Sistema per il monitoraggio e della verifica automatica delle procedure . . . . .	23
3.3.1	Gestione Dati . . . . .	24
3.3.2	Avvio del sistema . . . . .	26
3.3.3	Text-to-speech . . . . .	27
3.3.4	Modulo di formazione video per operatori . . . . .	28
3.3.5	Modulo gestione procedura . . . . .	30
3.4	Descrizione Interfaccia . . . . .	31
<b>4</b>	<b>Conclusioni e Lavori Futuri</b>	<b>36</b>
<b>5</b>	<b>Bibliografia</b>	<b>37</b>

# 1 Introduzione

Negli ultimi anni, i rapidi progressi tecnologici hanno provocato profondi cambiamenti nel contesto economico, sociale e culturale, con un impatto significativo sul mondo del lavoro. Tutto ciò ha determinato una trasformazione nel modo in cui le aziende producono, distribuiscono e commercializzano i loro prodotti e servizi. L'introduzione di nuove tecnologie ha reso possibile lo sviluppo di soluzioni innovative per migliorare la sicurezza e la qualità del lavoro degli operatori. Tra le tecnologie emergenti, i dispositivi indossabili dotati di una videocamera (es. smart-glasses) come Microsoft HoloLens 2<sup>1</sup> (vedi Fig. 1), rappresentano uno strumento estremamente utile in ambito industriale. In particolare, questi dispositivi indossabili sono dotati di una camera che permette di osservare il mondo dal punto di vista dell'operatore che li indossa ed inoltre permette all'operatore di poter lavorare avendo le mani libere. Inoltre, la maggior parte di questi dispositivi permettere di utilizzare la realtà aumentata per mostrare a schermo informazioni aggiuntive sull'ambiente e sugli oggetti presenti. Al giorno d'oggi esistono varie soluzioni che sono state già sviluppate per fornire servizi di supporto per gli operatori ma che sono passive, ovvero sono statiche e non sono influenzate dal comportamento dell'essere umano (es. assistenza remota, visualizzazione statica in realtà aumentata di informazioni). Lo scopo di questa tesi è quello di sviluppare un servizio "attivo" che sia in grado di supportare il lavoratore comprendendo automaticamente il suo comportamento dall'analisi di immagini e video acquisite da una camera. Un possibile impiego dei dispositivi indossabili è la registrazione di video delle attività degli operatori in modo pratico e meno invasivo rispetto alle videocamere tradizionali. Questi video possono essere successivamente elaborati da algoritmi di machine learning e Computer Vision al fine di supportare l'operatore. Tra gli algoritmi di computer vision applicabili, gli algoritmi di Human-Object interaction (HOI) detection sono particolarmente utili in campo industriale. Questi algoritmi sono in grado di rilevare le interazioni tra esseri umani e oggetti, fornendo le basi per lo sviluppo di vari servizi a supporto del lavoratore. Ad esempio, possono essere impiegati per creare sistemi di allerta in grado di notificare l'operatore quando sta interagendo con un oggetto potenzialmente pericoloso (come le schede ad alta tensione) o per fornire dati statistici sul tempo di utilizzo degli strumenti di lavoro. Inoltre possono essere utilizzati in comunicazione con dispositivi IoT per poter ac-

---

<sup>1</sup><https://www.microsoft.com/it-it/hololens/hardware>

cendere o spegnere in automatico vari dispositivi o macchinari solamente quando l’utente li sta utilizzando e possono anche essere utilizzati per la formazione dei nuovi operatori mostrando video su come si utilizza un determinato oggetto o come si effettua uno specifico passo di una procedura di manutenzione.



Figure 1: Dispositivo di realtà mista *Microsoft HoloLens 2*

Il software presentato in questa tesi rappresenta uno dei numerosi casi d’uso possibili di un algoritmo di HOI detection. L’obiettivo del sistema sviluppato è quello di monitorare e verificare il corretto svolgimento di una procedura industriale (es. una procedura di manutenzione), basandosi sulle interazioni che avvengono tra l’utente e gli oggetti coinvolti nel processo produttivo. In particolare, data una procedura di installazione o manutenzione il sistema guida l’operatore durante ogni passaggio mediante una combinazione di segnali audio e video in tempo reale, al fine di assicurare che tutte le fasi della procedura siano eseguite correttamente e in sicurezza. Questo è possibile tramite l’analisi di immagini e video acquisiti dalla camera di un paio di smartglasses o una camera installata su un caschetto da lavoro. La Figura 2 mostra il sistema sviluppato.

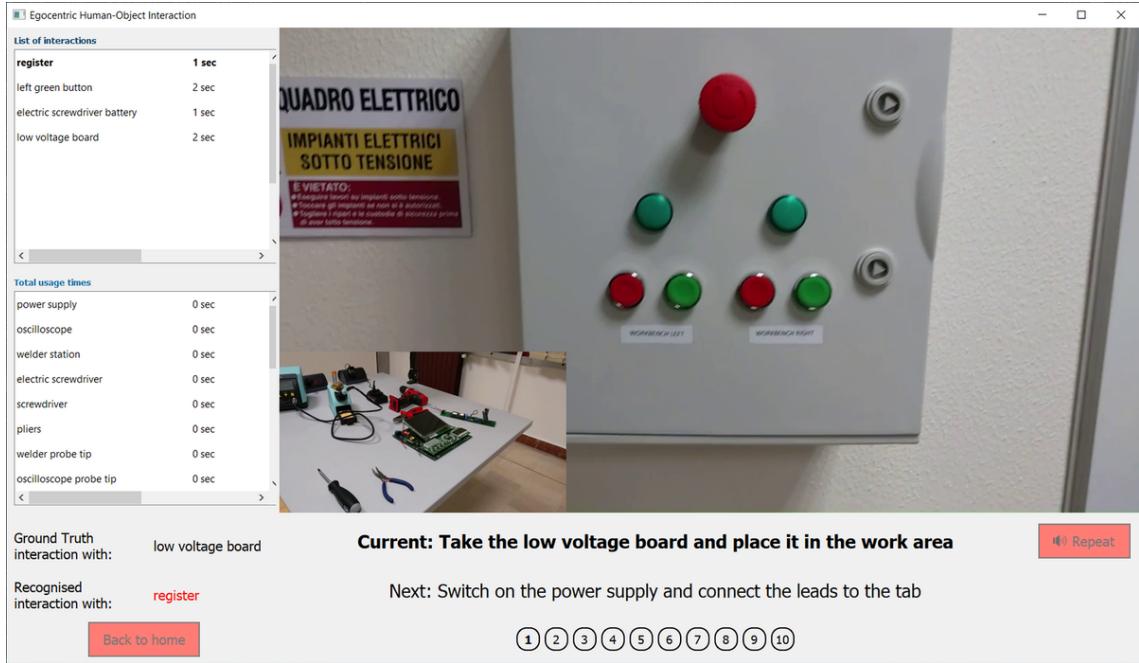


Figure 2: Sistema sviluppato per il monitoraggio e la verifica di una procedura industriale data una procedura di installazione o manutenzione

La struttura della presente tesi è la seguente:

- nel Capitolo 2 verranno descritti i principali algoritmi di riconoscimento di HOI, sia dal punto di vista della terza persona che da quello della prima persona. Inoltre, verranno analizzati brevemente alcuni task correlatati a quello di riferimento, come ad esempio gli algoritmi di object detection o riconoscimento delle mani. Infine, si fornirà una panoramica delle principali applicazioni esistenti per il supporto di operatori nel dominio industriale.
- nel Capitolo 3 verranno analizzate le tecnologie adoperate e le caratteristiche implementative del sistema sviluppato, descrivendo dettagliatamente l’interfaccia grafica e le funzionalità implementate;
- Infine, nel Capitolo 4, si discuteranno brevemente i possibili sviluppi futuri di questo lavoro.

## 2 Related Work

### 2.1 Human-Object Interaction detection

L'Human-Object Interaction detection (HOI) è un task studiato nel campo della Computer Vision il cui obiettivo è quello di riconoscere e comprendere le interazioni tra gli esseri umani e gli oggetti presenti in un ambiente.

Data un'immagine, gli algoritmi di HOI Detection: 1) individuano l'essere umano nella scena, 2) individuano con quali oggetti sta interagendo l'umano e 3) predicono un verbo che descrive l'interazione tra l'uomo e gli oggetti Figure 3.

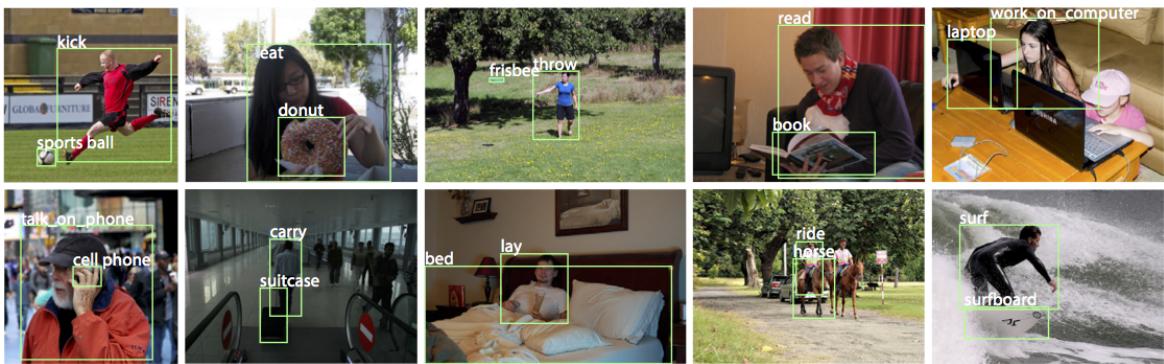


Figure 3: Esempio di HOI Detection

### **2.1.1 Algoritmi di Human-Object Interaction detection da Third Person point of View**

Le tecnologie di rilevamento e analisi di immagini e video in Third Person View (TPV) sono utilizzate in diverse applicazioni. Come ad esempio, il rilevamento di pedoni per i sistemi di assistenza alla guida [7] o la realtà aumentata [18] per la comprensione della scena e delle attività umane coinvolte, al fine di fornire all’utente un’esperienza interattiva con il mondo virtuale.

Inoltre, questa prospettiva permette una visione ampia e completa della scena permettendo di raccogliere informazioni dettagliate e complete su ciò che sta accadendo in un determinato contesto. Infatti risulta particolarmente utile per le applicazioni di video sorveglianza e monitoraggio [10], in cui le telecamere possono essere posizionate in posizioni discrete al fine di rilevare comportamenti sospetti e prevenire atti criminosi.

Nel caso di telecamere in grado di cambiare posizione o orientamento, un possibile approccio è la percezione attiva proposta da [23]. Esso implementa un sistema in grado di spostare la telecamera attraverso una sequenza di decisioni al fine di ottenere una visione completa dell’oggetto in esame. L’utilizzo della percezione attiva può ridurre significativamente l’influenza dell’ambiguità e dell’incertezza di una singola immagine, migliorando così le prestazioni.

Esistono diversi lavori in TPV per la comprensione delle interazioni uomo-oggetto, ad esempio:

- il lavoro presentato in [15] propone una Parallel Point Detection and Matching framework per il rilevamento delle HOI e le definisce come una tripletta di punti <punto umano, punto di interazione, punto dell’oggetto>, dove questi punti rappresentano il centro dei relativi bounding box.
- gli autori di [8] propongono un sistema di apprendimento multitask per affrontare il rilevamento delle HOI, il quale si compone di un ramo di rilevamento degli oggetti, un ramo relativo all’uomo e un ramo di interazione. Questo metodo si avvale di Region-based Convolutional Neural Network (R-CNN)

per generare regioni di interesse nell’immagine. Successivamente, il modello genera le associazioni tra le regioni contenenti persone e oggetti, permettendo di classificare il tipo di interazione.

- il lavoro di [26] propone un approccio completamente convoluzionale che individua direttamente le interazioni tra coppie uomo-oggetto. Predicendo i punti di interazione, che localizzano e classificano direttamente le interazioni.

In questo lavoro di tesi, abbiamo trattato il task di HOI Detection dal punto di vista della prima persona.

## 2.2 Algoritmi di Human-Object Interaction da First Person (Egocentric) View

Negli ultimi anni, grazie ai rapidi progressi tecnologici dei dispositivi indossabili, l’interesse per la percezione umana in prima persona (FPV), nota anche come “visione egocentrica”, è cresciuto notevolmente.

Questo ha portato allo sviluppo di diversi applicativi innovativi con l’obbiettivo di migliorare la qualità della vita delle persone. Ad esempio, nel campo medico, è stato proposto un metodo in grado di riassumere un video egocentrico [25] che può migliorare la salute dei pazienti affetti da Alzheimer, generando video riassuntivi egocentrici di persone, oggetti e farmaci importanti per facilitare il richiamo dei loro ricordi dimenticati.

Uno dei benefici dell’utilizzo di dispositivi di visione in prima persona (FPV) come action camera e smart-glasses è la capacità di acquisire immagini e video che forniscono una prospettiva egocentrica. Questo punto di vista consente di avvicinarsi all’esperienza visiva dell’utente che utilizza tali dispositivi e di rilevare le parti più significative della scena, come le mani durante le interazioni dell’utente.

Quest’ultime rappresentano il principale canale di interazione dell’essere umano con l’ambiente circostante, poiché consentono di manipolare oggetti, percepire il mondo esterno e comunicare con gli altri. Inoltre, la loro posizione e configurazione riflette l’attività svolta rendendole degli importanti indicatori all’interno degli algoritmi di machine learning.

Nonostante i vantaggi questo tipo di visione comporta anche una serie di sfide da affrontare:

- **Condizioni di illuminazione** i luoghi in cui vengono acquisiti i video sono variabili e incontrollabili, ad esempio, visitare un luogo turistico durante una giornata di sole, guidare un’auto di notte, preparare il caffè in cucina;
- **Requisiti in tempo reale** la visione egocentrica è molto utile in applicazioni da usare in real time, pertanto i sistemi devono essere in grado di elaborare e

analizzare i dati in modo rapido ed efficiente al fine di supportare attività in tempo reale.

- **Peggior qualità del contenuto:** a causa del movimento della testa o del torace, i dispositivi indossabili tendono a produrre video con molti segmenti sfocati e traballanti. Inoltre, l'inclinazione della testa durante la registrazione può influenzare l'angolazione della ripresa e causare inclinazioni nell'immagine risultante. Questo a differenza di notevoli applicazioni TPV, in cui il cameraman cerca di stabilizzare la registrazione o la telecamera è fissata.

La prospettiva in prima persona risulta vantaggiosa per la realizzazione di algoritmi di rilevamento delle interazioni tra esseri umani e oggetti (EHOI detection), in quanto consente di acquisire una comprensione più approfondita delle azioni umane e del loro contesto.

Il task di HOI Detection si articola in diverse attività:

- **Object detection** Il processo di rilevamento e riconoscimento degli oggetti si concentra sull'individuazione e la classificazione degli oggetti presenti in un'immagine o in un video. L'obiettivo principale è quello di fornire in output il rettangolo di selezione che delimita l'oggetto rilevato e la label della sua classe di appartenenza.

Vi sono numerose applicazioni che mirano alla rilevazione di oggetti specifici all'interno di immagini o video. Tra questi, due esempi comuni in terza persona sono la rilevazione dei volti Figure 4 e dei pedoni Figure 5. La prima consiste nell'individuare la presenza di uno o più volti in un'immagine o in un video, mentre la seconda ha l'obiettivo di individuare la presenza di persone all'interno di una scena.



Figure 4: Rilevazione volti



Figure 5: Rilevazione pedoni

Uno dei metodi più efficaci per eseguire questo task è Faster R-CNN (Faster Region-based Convolutional Neural Network)[22]. Questo metodo, come YOLO (You Only Look Once) [21] e SSD (Single Shot Detector) [16], si basa sull'utilizzo di una rete neurale convoluzionale (CNN) per individuare gli oggetti presenti in un'immagine.

Uno dei principali problemi consiste nel fatto che gli oggetti possono comparire in qualsiasi posizione dell'immagine, presentando rapporti di aspetto e dimensioni diversi.

- **Hand detection** Il rilevamento della mano mira a identificare le mani in un'immagine o in un video.

Quando una persona afferra un oggetto, la mano spesso ne copre la maggior parte occludendolo e rendendo complesso il rilevamento dell'oggetto stesso. Tuttavia, l'aspetto della mano può fornire importanti informazioni sulla posizione, la forma, le dimensioni e la posa dell'oggetto attivo.

Le tecniche più comuni utilizzate per il rilevamento delle mani includono:

- **il rilevamento basato su feature:** che si basa su tecniche di feature extraction rispetto la forma, la texture e il colore delle mani, come nel

paper [14];

- **il rilevamento basato su Deep Learning:** che utilizza deep neural network per identificare le mani in una scena. Come ad esempio, la rete proposta da [11] che apprende le caratteristiche delle mani dalle immagini e le utilizza per il loro riconoscimento;
- **il rilevamento basato su modelli 3D:** che utilizza sensori come telecamere stereo o sensori di profondità per acquisire informazioni 3D della scena e rilevare le mani [9].

Un aspetto importante da considerare è che le mani sono spesso coinvolte in contesti complessi, in cui sono presenti molte persone o una singola persona che svolge diverse attività contemporaneamente, rappresenta una sfida significativa per questa attività.

Grazie ai continui progressi dei dispositivi indossabili, sono stati proposti diversi dataset di immagini e video catturati dal punto di vista della prima persona, che rappresentano una fonte preziosa di dati per l'addestramento degli algoritmi di EHOI.

In questo contesto il dataset MECCANO, rappresenta il primo dataset di video egocentrici a studiare le interazioni uomo-oggetto in ambienti di tipo industriale [20]. Il dataset è stato acquisito da 20 partecipanti ai quali è stato chiesto di costruire un modello di moto, per il quale dovevano interagire con piccoli oggetti e strumenti.

Il suo successore [19] è un set di dati multimodale di video egocentrici, creato con lo scopo di studiare la comprensione del comportamento umano in ambienti industriali. La multimodalità è caratterizzata dalla presenza di segnali di sguardo, mappe di profondità e video RGB acquisiti simultaneamente con un casco personalizzato.

Inoltre Ego4D [12] è un set di dati egocentrici su larga scala che offre 3.670 ore di video di attività quotidiane che abbracciano centinaia di scenari (casa, esterno, luogo di lavoro, tempo libero, ecc.) ripresi da 931 persone che indossano una telecamera, provenienti da 74 località del mondo e da 9 Paesi diversi.

Inoltre gli autori di [4] hanno considerato l'uso di dati sintetici per il riconoscimento delle interazioni uomo-oggetto

Nonostante la diffusione dei dispositivi indossabili, nel mercato attuale, le applicazioni si limitano principalmente a offrire servizi di assistenza remota con un operatore o servizi di realtà aumentata passiva.

Tuttavia, questo lavoro propone un servizio di supporto per i lavoratori che attivamente comprende il comportamento umano attraverso l'analisi di immagini catturate da una telecamera.

Il compito di EHOI, secondo gli autori di [19], consiste nella produzione di coppie <verbo, oggetto>. L'articolo ha analizzato il problema del riconoscimento di oggetti attivi in ambienti di tipo industriale senza considerare le mani.

Esistono diversi approcci per l'EHOI detection, i precedenti lavori si sono concentrati sul rilevamento delle mani che interagiscono su un oggetto senza riconoscerlo [2, 17]. Altri lavori recenti si sono concentrati sul rilevamento di EHOI agnostici rispetto alle classi di oggetti [5].

### **2.2.1 Egocentric Human-Object Interaction Detection Exploiting Synthetic Data**

Nel lavoro [13] si affronta l'attività di rilevamento degli EHOI in un dominio industriale e si indaga su l'utilità dell'utilizzo di dati sintetici per la formazione quando il sistema deve essere testato su dati reali. Il modello implementato Figure 6 è quello utilizzato dal software di controllo e monitoraggio procedura.

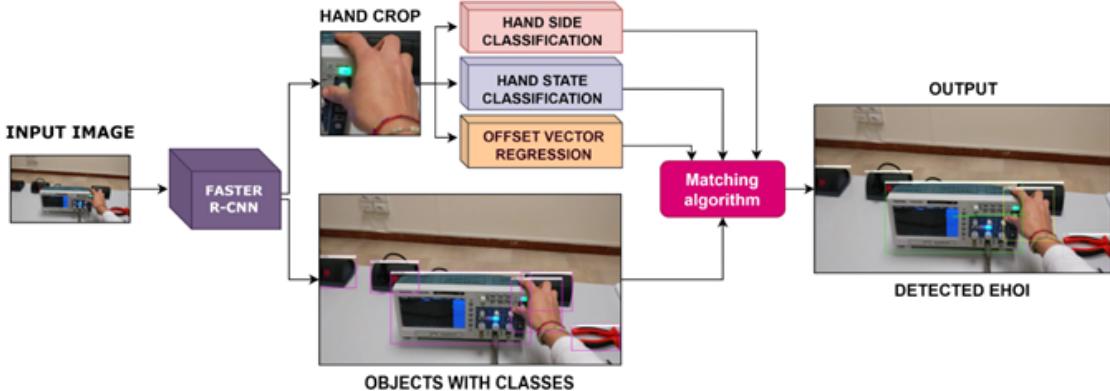


Figure 6: Modello per il rilevamento degli EHOI

L'approccio proposto predice per ciascun frame video:

- **hand** il riquadro di delimitazione della mano e il lato (sinistro/destro);
- **contact\_state** lo stato di contatto che può essere in contatto oppure non in contatto.
- **active\_object** il riquadro di delimitazione dell'oggetto attivo e la classe di appartenenza;
- **other\_objects** i riquadri di delimitazione degli oggetti non attivi e la classe di appartenenza.

Producendo in output:  $\langle hand, contactstate, activeobject, \langle otherobjects \rangle \rangle$ . La struttura del modello è composta da diversi moduli:

- **Hands and object detection** che utilizza la rete neurale *Faster R-CNN* per predirre una tupla  $(x, y, w, h, c)$  per ogni oggetto/mano nell'immagine, dove i valori di  $(x, y, w, h)$  rappresentano le coordinate del riquadro di delimitazione e  $c$  la classe dell'oggetto prevista.

Produce in output anche un crop dell'immagine nella sezione che contiene la mano, questa verrà passata ai moduli: hand side classification, hand state classification e offset vector regression;

- **Hand side classification module** composto da un *Multy-Layer-Perceptron (MLP)* con due strati completamente connessi, partendo dalle mani rilevate ne predice il lato.
- **Hand state classification module** composto da un MLP con due strati completamente connessi, partendo dalle mani rilevate, permette di stabilire lo stato di contatto della mano.
- **Offset vector regression module** seguendo l’approccio proposto in [8], produce in output un vettore offset che collega il centro di ogni bounding box della mano al centro del corrispondente oggetto attivo.

Tramite un MLP con due strati completamente connessi, produce in output la tripletta  $\langle v_x, v_y, m \rangle$ , dove  $v_x$  e  $v_y$  rappresentano le componenti del versore  $v$  ed  $m$  la magnitudine;

- **Matching algorithm** prende in input i risultati dei moduli precedenti e calcola il centro previsto dell’oggetto attivo per ogni mano in contatto con un oggetto, rappresentato dal punto dell’immagine  $p_{interaction}$ .

Per selezionare l’oggetto attivo, l’algoritmo considera il rettangolo di selezione dell’oggetto il cui centro è più vicino al  $p_{interaction}$  dedotto. Inoltre, viene effettuato un controllo per verificare se il rettangolo di delimitazione dell’oggetto ha un’intersezione non nulla con il rettangolo di delimitazione della mano.

## **2.3 Applicazioni esistenti a supporto degli operatori nel dominio industriale**

La quarta rivoluzione industriale, nota come "Industria 4.0", sta portando una profonda trasformazione nel modo in cui le aziende producono, distribuiscono e commercializzano i loro prodotti e servizi. Questa trasformazione è resa possibile grazie all'integrazione di tecnologie avanzate, come l'IOT, il cloud computing, l'analisi dei dati, l'Intelligenza Artificiale, il Machine Learning e la produzione in cloud.

L'utilizzo di queste tecnologie consente di creare un ambiente di lavoro più sicuro, migliorare la produttività e ridurre gli errori umani. Ad esempio, l'uso di sistemi IoT consente di monitorare in tempo reale la produzione e controllarla, riducendo il rischio di errori e aumentando l'efficienza del processo produttivo. Inoltre, il cloud computing offre l'opportunità di archiviare e accedere ai dati da qualsiasi luogo, migliorando l'efficienza e la flessibilità dei processi produttivi e dei sistemi di gestione aziendale.

In questo contesto sono stati realizzati diversi sistemi a supporto dell'operatore, come:

- **Sistemi di assistenza alla produzione:** questi strumenti sono progettati per assistere gli operatori durante la produzione, ad esempio fornendo istruzioni dettagliate su come svolgere determinate attività o aiutando a gestire le attrezzature.

Ad esempio, il paper [3] discute l'importanza dei sistemi di assistenza industriale per migliorare l'interazione uomo-macchina e le capacità degli operatori. Inoltre fornisce un'analisi dettagliata di diverse tecnologie di assistenza industriale, tra cui la realtà aumentata, la realtà virtuale, i dispositivi portatili, i sensori e l'Internet delle cose (IoT), e discute come queste tecnologie possono migliorare l'efficienza e la sicurezza dei processi produttivi.

- **Sistemi di gestione della conoscenza:** questi strumenti consentono di catturare, condividere e utilizzare la conoscenza acquisita dagli operatori nel corso del tempo, migliorando la formazione e riducendo gli errori.

Ad esempio, l’obiettivo di questo articolo [6] è quello di analizzare come la gestione della conoscenza (KM) possa fornire supporto all’attuazione dell’Industria 4.0. I KM mirano catturare, conservare, condividere e riutilizzare le conoscenze create dai lavoratori durante le attività di routine per migliorare i processi produttivi.

- **Sistemi di realtà aumentata:** questi sistemi utilizzano tecnologie di realtà virtuale o aumentata per fornire agli operatori informazioni dettagliate sui processi di produzione, come ad esempio la posizione esatta degli strumenti e delle parti da assemblare.

Ad esempio, l’articolo [1] si concentra sull’applicazione dell’AR in industrie come la costruzione navale, lo shopping online, la chirurgia e l’istruzione. L’utilizzo di questa tecnologia permette di visualizzare informazioni digitali sovrapposte al mondo reale, come ad esempio dati di produzione o istruzioni di montaggio migliorando la produttività e la qualità del lavoro.

Tra le tecnologie innovative introdotte, gli algoritmi per il rilevamento delle EHOI in ambito industriale offrono numerosi vantaggi, tra cui la capacità di anticipare le interazioni tra gli oggetti e gli operatori. Questo può contribuire a migliorare la sicurezza in fabbrica, ad esempio segnalando tempestivamente all’operatore un’azione o un’interazione potenzialmente pericolosa.

Altri possibili utilizzi riguardano il monitorare l’utilizzo delle macchine, programmare le operazioni di calibrazione, suggerire all’operatore come utilizzare una macchina o un oggetto specifico e emettere notifiche su azioni che potrebbero essere mancate in una pipeline di produzione [24]. In questo modo si ottimizzano le prestazioni delle macchine e si migliora l’efficienza del processo produttivo.

Inoltre, nel contesto industriale, gli oggetti di interesse sono già noti in anticipo, come ad esempio gli strumenti e gli utensili con cui l’utente deve interagire. Tuttavia, il rilevamento di tali interazioni in questo contesto risulta ancora poco sviluppato a causa della mancanza di dataset pubblici disponibili e delle questioni legate alla privacy e alla tutela dei segreti industriali.

## 3 Software Sviluppato

### 3.1 Tecnologie utilizzate

Il sistema per monitorare e verificare le procedure di lavoro nel contesto industriale proposto in questa tesi si basa sul linguaggio di programmazione Python e fa uso di diverse librerie per implementare le sue funzionalità. In particolare, la libreria *PyQT5*<sup>2</sup> è stata impiegata per la creazione e la gestione dell’interfaccia grafica del sistema, mentre la libreria *pyttsx3*<sup>3</sup> è stata utilizzata per fornire il supporto vocale all’operatore. Inoltre, la libreria *OpenCV*<sup>4</sup> è stata impiegata per l’acquisizione, l’elaborazione e la riproduzione di video.

#### 3.1.1 Libreria per la gestione delle interfacce grafiche

*PyQT5* è un set di collegamenti Python per *Qt v5*<sup>5</sup> che permette di creare interfacce grafiche su diverse piattaforme. Grazie ai *Widget*, ovvero i componenti dell’interfaccia utente, è possibile gestire tutti gli aspetti grafici di una finestra, sia statici che interattivi.

I componenti statici Figure 7 sono elementi grafici fissi che vengono principalmente utilizzati per mostrare informazioni all’utente, come ad esempio etichette o pulsanti. I componenti interattivi Figure 8, invece, cambiano in base all’interazione dell’utente con l’interfaccia. Ad esempio, un campo di input permette all’utente di digitare del testo e inviarlo all’applicazione. Per organizzare e gestire la dispo-

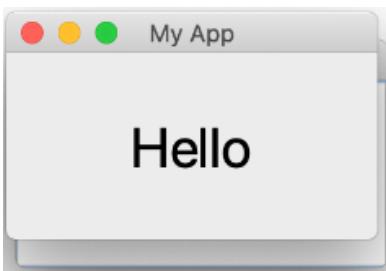


Figure 7: Esempio Widget statico

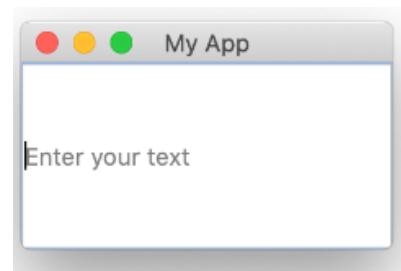


Figure 8: Esempio Widget dinamico

sizione dei *Widget* all’interno della finestra, *PyQT5* fornisce un sistema di *Layout*

<sup>2</sup>PyQt5, documentation: <https://doc.qt.io/qtforpython/>

<sup>3</sup>pyttsx3, documentation: <https://pyttsx3.readthedocs.io/en/latest/>

<sup>4</sup>OpenCV, documentation: <https://opencv.org/>

<sup>5</sup>Qt V5, documentation: <https://doc.qt.io/qt-5.15/>

responsive. Per utilizzare il *Layout* desiderato basta crearne un’istanza e assegnarla ad un *Widget* o ad un contenitore di *Widget*.

Per gestire gli eventi generati dall’interfaccia utente, come i click del mouse o le pressioni dei tasti, *PyQT5* permette di definire funzioni specifiche che vengono eseguite quando l’evento si verifica. In questo modo è possibile fornire una risposta personalizzata e controllata alle azioni dell’utente.

Inoltre, offre una vasta gamma di opzioni di personalizzazione per l’aspetto della Graphical User Interface (GIU), tra cui l’utilizzo di fogli di stile CSS e la definizione di temi grafici personalizzati.

Per migliorare le prestazioni delle applicazioni *PyQT5* e ottimizzare l’utilizzo delle risorse di sistema, è possibile sfruttare i thread, che consentono l’esecuzione contemporanea di più flussi di istruzioni. Grazie all’uso dei thread è possibile distribuire il carico di lavoro su più processi e gestire in modo più efficiente l’esecuzione di attività computazionalmente onerose. Questa tecnologia consente all’applicazione di evitare il blocco dell’interfaccia utente durante l’elaborazione dei dati, migliorando notevolmente la sua capacità di risposta.

Nelle applicazioni *PyQT5* il *Main thread*, conosciuto anche come *GUI thread*, costituisce il cuore dell’applicazione e si occupa di eseguire l’interfaccia grafica, mentre i *Worker threads* vengono creati al momento del bisogno per gestire le attività computazionalmente onerose.

Per la comunicazione tra thread e per la corretta gestione delle risorse condivise, è possibile utilizzare *segnali* e *slot* Figure 9. Questi strumenti consentono di sincronizzare l’elaborazione di dati tra più thread, evitando i problemi di concorrenza che possono causare errori e malfunzionamenti nel sistema.

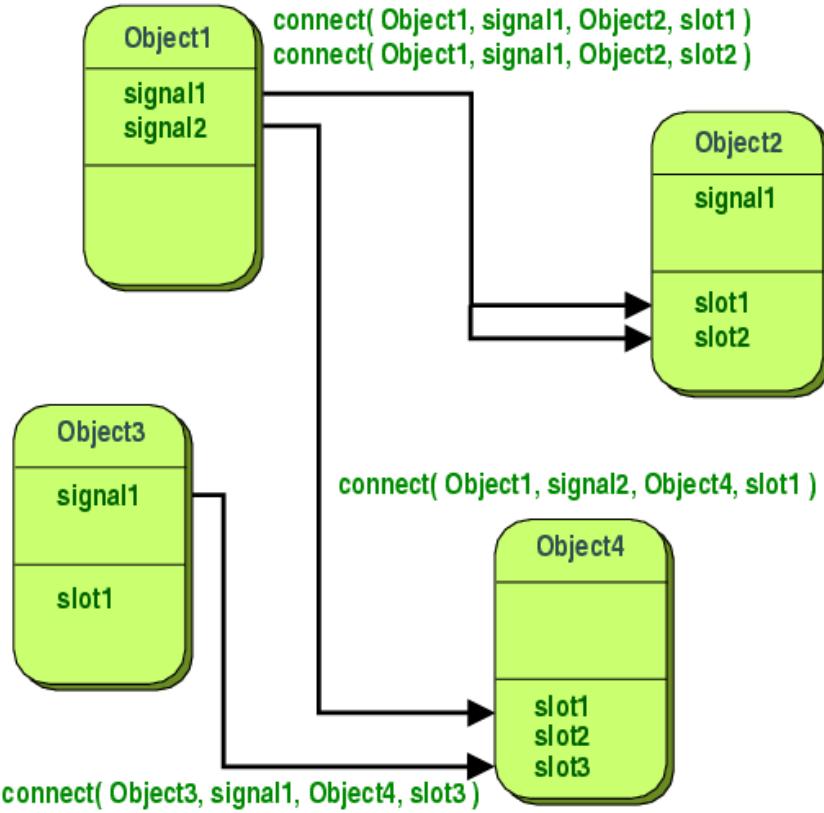


Figure 9: Quattro thread che comunicano inviando segnali e ricevendo tali segnali tramite gli slot. Ad esempio `connect( Object1, signal1, Object2, slot1 )` in questo caso sto connettendo `signal1` emesso da **Object1** allo `slot1` di **Object2**.

### 3.1.2 Libreria per la gestione del supporto vocale

Per soddisfare i requisiti del sistema sviluppato, è stata importata la libreria di supporto vocale pytsxs3. Questa libreria consente la riproduzione vocale di un testo ricevuto in input senza la necessità di una connessione Internet, un requisito funzionale importante per l'applicazione sviluppata.

Inoltre, pytsxs3 supporta diversi sintetizzatori vocali, il che consente di utilizzare la libreria su diverse piattaforme e offre diverse opzioni di configurazione per personalizzare la riproduzione vocale del testo, come la velocità di lettura, il volume della voce, il tipo di voce e il tasso di campionamento audio.

### 3.1.3 Libreria per la gestione delle immagini

*OpenCV* è una libreria open source per la computer vision e il machine learning, che mette a disposizione numerosi algoritmi per l'elaborazione di immagini e video.

*OpenCV* offre una vasta gamma di strumenti per effettuare trasformazioni come la conversione dei colori, il ridimensionamento e il ritaglio delle immagini. Tra le sue funzionalità vi è la capacità di manipolare i pixel delle immagini, individuare i bordi e acquisire ed elaborare i dati provenienti da fotocamere.

Inoltre, *OpenCV* è largamente impiegata per l’elaborazione e l’analisi di immagini e video, inclusa l’acquisizione di video catturati da videocamere. Grazie alle sue funzionalità e alla facile integrazione con i più comuni linguaggi di programmazione, questa libreria viene utilizzata in molti campi della computer vision.

## 3.2 Laboratorio ENIGMA

L'algoritmo di EHOI utilizzato nell'applicazione per riconoscere le interazioni uomo-oggetto è stato testato in un laboratorio Figure 10 che rappresenta uno scenario industriale realistico “ENIGMA” .



Figure 10: Laboratorio di tipo industriale “ENIGMA”

Nel laboratorio sono presenti 23 oggetti diversi sia fissi, come pannelli elettrici, un alimentatore e una stazione di saldatura, sia mobili, come cacciaviti e altri oggetti. Un esempio in Figure 11.



Figure 11: Oscilloscopio da diversi punti di vista

Inoltre, sono presenti diversi dispositivi IoT installati nelle prese del tavolo di lavoro e nel pannello elettrico che permettono di accendere e spegnere l'elettricità degli strumenti collegati alle prese.

### **3.3 Sistema per il monitoraggio e della verifica automatica delle procedure**

Questa tesi riguarda lo sviluppo di un sistema per il monitoraggio e la verifica di procedure di lavoro nel contesto industriale. Il sistema sviluppato guida l'operatore durante le diverse fasi di una procedura analizzando le immagini acquisite da una camera montata su un caschetto e riconoscendo le interazioni che avvengono tra il lavoratore e gli oggetti presenti nell'ambiente circostante.

Il riconoscimento dell'interazione uomo-oggetto corrente è affidato a un algoritmo di machine learning e Computer Vision appositamente progettato per riconoscere le interazioni uomo-oggetto dal punto di vista egocentrico nell'ambito industriale.

Per velocizzare il processo di sviluppo, le interazioni sono state generate casualmente utilizzando un generatore di interazioni casuale. Successivamente, si è passati ad una fase di integrazione con il modello di machine learning per il riconoscimento delle interazioni, in modo da ottenere un unico sistema in grado di assistere gli operatori durante le procedure di lavoro.

Il sistema permette all'utente di selezionare una specifica procedura dall'elenco delle procedure disponibili poiché, in un ambiente industriale, gli operatori possono dover eseguire diverse procedure a seconda dell'output finale del processo di lavoro.

Per memorizzare le informazioni sulle procedure, si utilizza un dizionario gerarchico in Python. Inoltre, al momento dell'avvio, vengono impostati dei parametri predefiniti per garantire che il sistema parta da uno stato coerente.

Il sistema è costituito da un insieme di moduli:

- **Text-to-speech:** viene richiamato automaticamente per far pronunciare a una voce la descrizione del passo corrente della procedura;
- **Modulo di formazione video per operatori:** riproduce un video rivolto agli operatori, che illustra la corretta esecuzione del passo corrente;

- **Modulo gestione procedura:** verifica che l'operatore stia interagendo con l'oggetto previsto per il passo corrente. In caso affermativo aggiorna i *Widget* e i parametri necessari per passare alla fase successiva.

### 3.3.1 Gestione Dati

Le informazioni relative alle procedure utilizzate dal sistema sono contenute in un file di input in formato JSON. Una volta acquisito, il contenuto del file viene trasferito in un dizionario Python che è organizzato in modo gerarchico.

La struttura del dizionario è la seguente:

```
dict[procedureId][stepId] = campi
```

Nello specifico, ogni procedura viene identificata da un intero univoco “**procedureId**”, utilizzato come chiave di primo livello nel dizionario. La chiave di secondo livello del dizionario è un intero “**stepId**” che indica il passo corrente della procedura.

Grazie a questa struttura di accesso, è possibile recuperare facilmente i campi relativi a ciascun passo della procedura selezionata dall’utente.

La struttura di ciascun passo della procedura è la seguente:

- **description:** descrizione testuale del passaggio attuale;
- **interactionId:** id interazione;
- **interaction:** nome interazione;
- **tutorialId:** id del video tutorial relativo all’interazione.

Di seguito una delle procedure del sistema.

```
1 {
2     "procedure_id": 1,
3     "procedure_steps": [
4         {
5             "step_id": 0,
6             "description": "Take the low voltage board and place it in the work area",
7             "interaction_id": 8,
8             "interaction": "low voltage board",
9             "tutorial_id": 0
10        },
11        {
12            "step_id": 1,
13            "description": "Take the electric screwdriver",
14            "interaction_id": 3,
15            "interaction": "electric screwdriver",
16            "tutorial_id": 1
17        },
18        {
19            "step_id": 2,
20            "description": "Adjust the power supply voltage knob by setting a voltage of
21                5 volts",
22            "interaction_id": 0,
23            "interaction": "power supply",
24            "tutorial_id": 2
25        },
26        {
27            "step_id": 3,
28            "description": "Press the green button of the oscilloscope",
29            "interaction_id": 1,
30            "interaction": "oscilloscope",
31            "tutorial_id": 3
32        },
33        {
34            "step_id": 4,
35            "description": "Take the low voltage board and place it outside of the work
36                area",
37            "interaction_id": 8,
38            "interaction": "low voltage board",
39            "tutorial_id": 4
40        },
41        {
42            "step_id": 5,
43            "description": "Take the high voltage board and place it in the work area",
44            "interaction_id": 9,
45            "interaction": "high voltage board",
46            "tutorial_id": 5
47        },
48        {
49            "step_id": 6,
50            "description": "Adjust the temperature of the welder station to 170 degrees",
```

```

49     "interaction_id": 2,
50     "interaction": "welder station",
51     "tutorial_id": 6
52 },
53 {
54     "step_id": 7,
55     "description": "Use the welder probe tip",
56     "interaction_id": 0,
57     "interaction": "welder probe tip",
58     "tutorial_id": 7
59 },
60 {
61     "step_id": 8,
62     "description": "Adjust the temperature of the welder station to 160 degrees",
63     "interaction_id": 2,
64     "interaction": "welder station",
65     "tutorial_id": 6
66 },
67 {
68     "step_id": 9,
69     "description": "Press the red button of the oscilloscope",
70     "interaction_id": 1,
71     "interaction": "oscilloscope",
72     "tutorial_id": 3
73 },
74 {
75     "step_id": 10,
76     "description": "Adjust the power supply voltage knob by setting a voltage of
77         0 volts",
78     "interaction_id": 0,
79     "interaction": "power supply",
80     "tutorial_id": 2
81 },
82 {
83     "step_id": 11,
84     "description": "Take the high voltage board and place it outside the work
85         area",
86     "interaction_id": 9,
87     "interaction": "high voltage board",
88     "tutorial_id": 5
89 }

```

### 3.3.2 Avvio del sistema

All'avvio, vengono impostati i parametri di default, al fine di garantire che il sistema parta da uno stato coerente.

In particolare:

- il parametro procedureId viene impostato a 0, poiché, nel caso in cui l’utente non selezioni una procedura diversa dal pannello, di default viene selezionata la prima;
- lo stepId viene inizializzato a 0 poiché rappresenta il primo passo della procedura;
- il modulo di gestione procedura viene disattivato, poichè in questa fase non è necessario.

Dopo che l’operatore ha selezionato la procedura da eseguire, è necessario memorizzare il “procedureId” della procedura. Questo campo rappresenta un elemento essenziale per accedere alle informazioni relative alla procedura selezionata e recuperare i parametri necessari per l’esecuzione della stessa. A questo punto è possibile avviare il controllore automatico di procedura premendo il pulsante “start”.

Una volta avviato, il sistema accede ai campi relativi alla procedura selezionata tramite l’identificatore univoco e li memorizza in un nuovo dizionario.

```
actualProcedure = allProcedures[parameters["procedureId"]]
```

Inoltre, al fine di settare la procedura allo stato iniziale, i parametri relativi l’interazione di ground truth, la descrizione del passaggio attuale e del successivo vengono settati con i campi del primo passo della procedura selezionata.

Una volta che i parametri interni sono stati impostati correttamente, il sistema riproduce la descrizione del primo passo e mostra il relativo video di training per mostrare all’operatore l’attuale passo della procedura.

### 3.3.3 Text-to-speech

Il Text-to-speech è implementato all’interno di un *Worker* thread *PyQT5* e viene istanziato all’interno del thread principale, passando come argomenti i parametri

relativi alla procedura attualmente selezionata:

```
voice = Voice(actual_procedure)
```

Il metodo viene richiamato automaticamente al fine di far pronunciare ad una voce la descrizione del passaggio corrente quando l’interazione dell’operatore con l’oggetto viene riconosciuta come quella di “Ground truth”. Se l’interazione viene riconosciuta come tale, il sistema passa automaticamente al passaggio successivo della procedura.

Per avviare l’esecuzione del thread si richiama il metodo start():

```
voice.start()
```

Questa chiamata consente l’esecuzione parallela del metodo “run()”, il quale, utilizzando la funzione “.speak()” di Pytsx3, consente all’engine di pronunciare la descrizione del passaggio corrente della procedura, senza interferire con il thread principale:

```
pytsx3.speak(actionsId[step_count]["description"])
```

Quando l’engine termina viene inviato un segnale *PyQT5* “finished” che viene intercettato dal main thread e richiama il metodo che si occupa di settare le informazioni relative al passaggio successivo della procedura:

```
voice.finished.connect(set_next_procedure_step)
```

### 3.3.4 Modulo di formazione video per operatori

Per garantire una formazione adeguata all’operatore, oltre al supporto audio, all’avvio e ogni volta che si raggiunge un nuovo passo della procedura, viene riprodotto un video tutorial che mostra la corretta esecuzione del passaggio corrente. Figure 12



Figure 12: In basso a sinistra il video tutorial

Il video viene gestito all'interno di un thread PyQt, che permette la visualizzazione fluida dei differenti frame. La selezione del video da riprodurre avviene sulla base dell'ID del video tutorial relativo al passo attuale della procedura.

Quando è necessario avviare la riproduzione del video viene richiamato il metodo “startVideo” nel relativo thread:

```
acquisition_thread.start_video(parameters["videos_tutorial"])
```

All'interno del metodo, viene generato il percorso del file che contiene il video di training da riprodurre. Questa operazione viene eseguita utilizzando l'identificativo (ID) univoco del tutorial selezionato e la sua posizione nella directory di lavoro:

```
_training_video_path = "./data/training_videos/" + str(tutorial_id) + ".mp4"
```

Successivamente, il video viene letto utilizzando la funzione *VideoCapture* del pacchetto *OpenCV*, che genera un oggetto che lo rappresenta:

```
_training_video_cap = cv2.VideoCapture(_training_video_path)
```

Infine si avvia la riproduzione che avviene frame by frame. Ciascun frame viene ridimensionato utilizzando il metodo “`resize()`” del pacchetto *imutils* e memorizzato in una variabile. Questa viene passata alla finestra principale all’interno di un segnale per poter mostrare il frame a video nell’interfaccia utente.

Dopo la riproduzione dell’ultimo frame il video viene riavviato.

### 3.3.5 Modulo gestione procedura

Dopo che il modulo vocale ha finito di riprodurre il primo passaggio della procedura, il modulo di gestione procedura viene attivato. Il sistema verifica che l’interazione relativa all’attuale passaggio della procedura corrisponda all’interazione di “ground truth”, al fine di poter procedere al passaggio successivo.

Grazie a questa caratteristica, il software è in grado di automatizzare il processo di avanzamento nella procedura sulla base dell’interazione riconosciuta.

Quando si verifica la necessità di passare al passo seguente della procedura il sistema:

- richiama il Text-to-speech per produrre l’audio del passaggio da svolgere;
- aggiorna i parametri relativi l’interazione di ground truth, la descrizione del passo attuale e del successivo;
- aggiorna i *Widget* grafici;
- fa partire il successivo video tutorial;

### 3.4 Descrizione Interfaccia

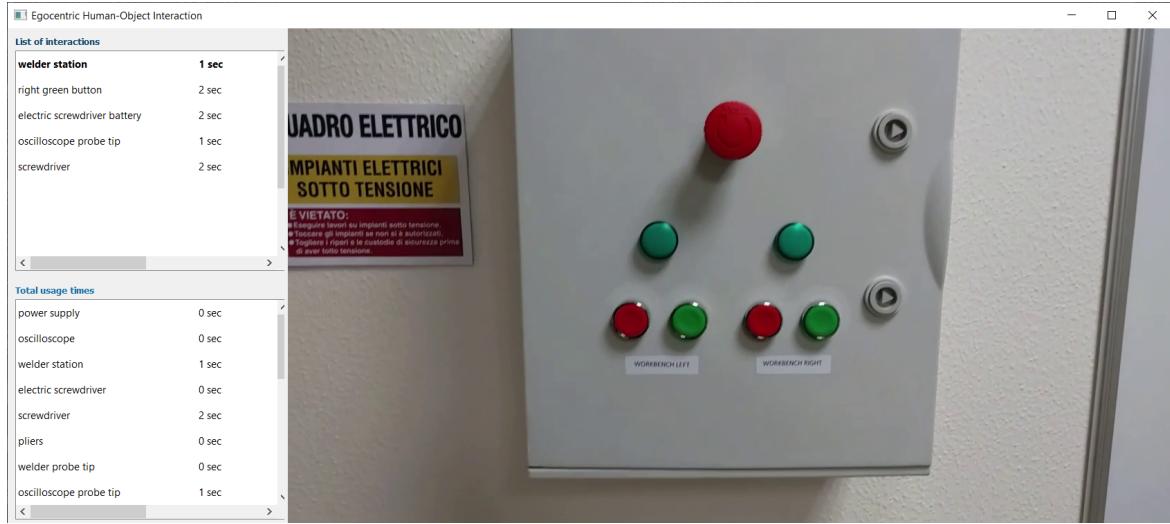


Figure 13: l’interfaccia con l’immagine corrente e due pannelli laterali che mostrano la lista delle interazioni e il tempo totale di utilizzo

Il sistema per il controllo delle procedure è stato implementato partendo da un’interfaccia preesistente Figure 13 , composta da:

- una finestra principale che mostra l’immagine corrente;
- un pannello laterale sinistro che visualizza la lista delle interazioni e il tempo totale di utilizzo degli strumenti impiegati.

I diversi *Widget* relativi al sistema sviluppato sono situati in una finestra *QDockWidget* posizionata nella parte inferiore dell’interfaccia utente. *QDockWidget* è una classe del framework *Qt* che rappresenta una finestra di dialogo fluttuante e ridimensionabile che può essere ancorata a diverse aree dell’interfaccia utente.

Le posizioni dei *Widget* a livello grafico sono gestite tramite l’utilizzo di un *Layout Qt* chiamato *QGridLayout*, il quale permette di organizzare i *Widget* all’interno di una griglia di righe e colonne.

L’interfaccia utente all’avvio dell’applicazione dà la possibilità all’utente di selezionare, attraverso un menu a tendina Figure 14, la procedura che si desidera eseguire. Quando si effettua una scelta, si può procedere all’avvio della procedura cliccando sul pulsante “start”.

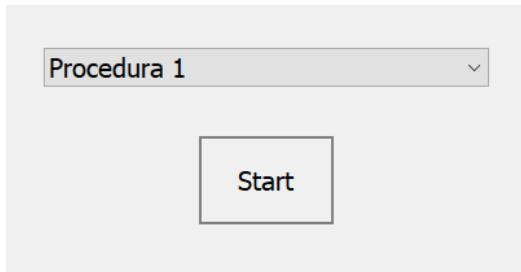


Figure 14: Pannello di selezione procedura

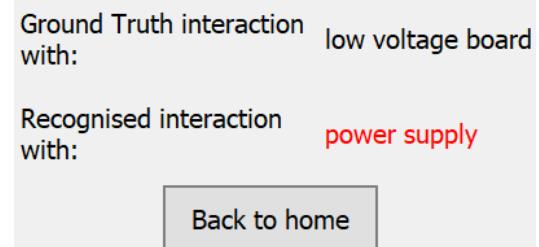


Figure 15: Pannello che mostra interazione di ground truth e interazione attualmente riconosciuta

Dopo aver avviato il software, il video tutorial viene riprodotto utilizzando la libreria *OpenCV* e mostrato nell’area sottostante al video principale Figure 16.

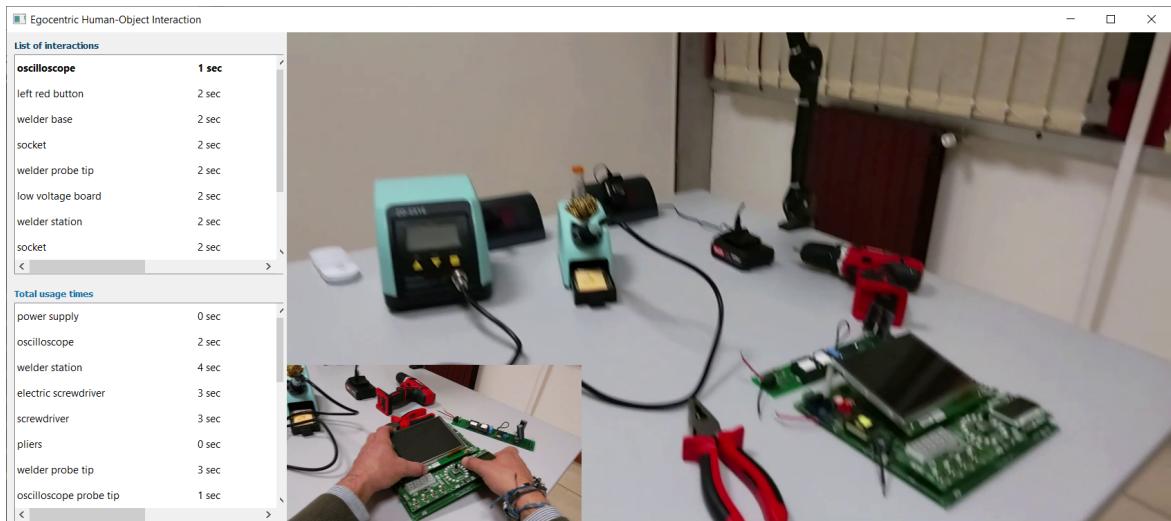


Figure 16: Nell’area sottostante al video principale il video tutorial

L’immagine viene visualizzata all’interno di una label utilizzando il metodo *Qt* “*setPixmap*”. Questo processo consente di elaborare e visualizzare il video in tempo reale, fornendo un’esperienza di apprendimento interattiva e immediata.

Inoltre, vengono attivati due pannelli che mostrano passo passo le informazioni relative al passaggio corrente della procedura e l’oggetto attualmente riconosciuto.

Un il pannello posto sul lato sinistro Figure 15 mostra l’interazione di ground truth e l’interazione attualmente riconosciuta dal sistema al fine di aiutare l’operatore a

monitorare la corretta esecuzione della procedura.

Inoltre presenta un tasto “Back to home” che riporta l’utente al menu di selezione della procedura, ripristina i valori di default dei parametri del sistema e gestisce la chiusura del video tutorial in caso di click del pulsante da parte dell’utente durante la riproduzione attiva.

Il pannello posto sul lato destro Figure 17 permette di visualizzare la procedura in corso ed è composto da composto da:

- in alto a sinistra una label che mostra la descrizione del passaggio corrente;
- in alto a destra un tasto con la scritta “Repeat” che permette all’utente di far ripetere alla voce la descrizione del passaggio;
- al centro una *Label* che mostra la descrizione del passaggio successivo;
- in basso un insieme di nodi che rappresentano il numero di passaggi previsti nella procedura. Il sistema crea un *QLabel* per ogni passo della procedura, il cui testo viene impostato ad una stringa numerica che rappresenta l’indice del passo nella lista.

**Current: Take the low voltage board and place it in the work area**



Next: Switch on the power supply and connect the leads to the tab

① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩

Figure 17: Pannello che mostra le informazioni relative all’attuale passaggio della procedura

Durante l’utilizzo dell’applicazione, quando l’interazione di “Ground truth” corrisponde all’interazione corrente, il sistema fornisce un feedback visivo per l’utente Figure 18:

- Il testo della label “Current” viene evidenziato in verde per indicare che il passo della procedura è stato completato correttamente;

- Il testo della label “Next” viene visualizzata in grassetto per indicare che è il passo da seguire attualmente;
- Il nodo relativo al “Current” viene colorato in verde, mentre il nodo relativo al “Next” viene visualizzato in grassetto per aiutare l’utente a comprendere a che passo della procedura si trova.

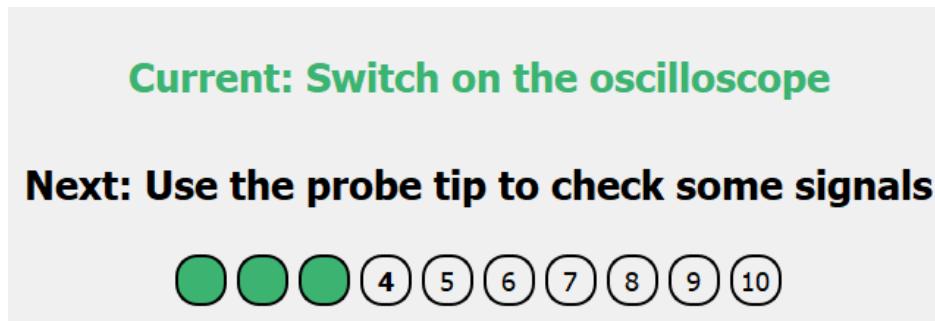


Figure 18: Aggiornamento del pannello quando l’interazione di Ground truth corrisponde a quella predetta dal modello.

- Nel pannello a sinistra Figure 19 la *Label* relativa all’interazione riconosciuta attualmente viene colorata di verde per indicare che corrisponde a quella di “Ground truth”.

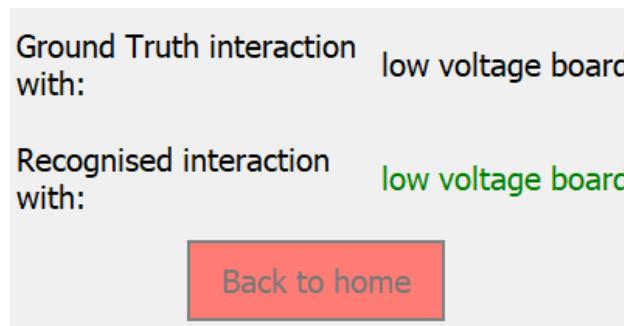


Figure 19: Interazione corrente e di Ground truth che corrispondono

Al termine della procedura oppure al click del tasto “Back to home”, i *Widget* preesistenti vengono nascosti in modo da poter essere modificati dinamicamente all’avvio di una nuova procedura selezionata.

Le funzioni che gestiscono lo stile dell’interfaccia utente sono state raggrup-

pate all'interno del file “style.py” e successivamente importate nel file principale dell'interfaccia utente, al fine di mantenere il codice organizzato e facilmente gestibile. Queste funzioni gestiscono tutti gli aspetti grafici dinamici del software, tra cui la visualizzazione o la scomparsa di pannelli specifici e la modifica del colore del testo delle label.

## 4 Conclusioni e Lavori Futuri

In questa tesi, è stato presentato un sistema di monitoraggio e verifica delle procedure industriali, il cui obiettivo è assistere l'operatore durante tutte le fasi di esecuzione di una procedura industriale a partire dall'analisi di immagini e video acquisite da una camera indossabile (es. smartglasses). Il sistema sfrutta un algoritmo di HOI detection per identificare l'oggetto con cui l'operatore sta interagendo e, in base alle informazioni rilevate, procedere automaticamente al passaggio successivo della procedura. L'obiettivo principale del sistema proposto è fornire supporto agli operatori tramite la visualizzazione di istruzioni vocali e visive relative al passaggio della procedura da svolgere. In questo modo, l'operatore avrà a disposizione le informazioni necessarie per eseguire la procedura correttamente, riducendo i rischi associati alle attività lavorative.

Le funzionalità implementate hanno dimostrato di essere efficaci nel facilitare l'esecuzione delle procedure di lavoro, ma ulteriori sviluppi e miglioramenti sarebbero opportuni per rendere il sistema più preciso rispetto i vari passaggi della procedura. Un esempio è l'integrazione un sistema di Optical Character Recognition (OCR) al fine di riconoscere testi dai video acquisiti. Ad esempio, questa sistema potrebbe essere utilizzato per verificare che l'operatore imposti correttamente i settaggi di uno strumento (per esempio, la temperatura di un saldatore o il voltaggio di un alimentatore). Inoltre, per migliorare l'esperienza dell'utente, si potrebbe considerare l'implementazione di nuove funzionalità come un sistema di riconoscimento vocale per poter interagire con l'applicazione senza l'uso delle mani. Per valutare l'efficacia del sistema e identificare possibili aree di miglioramento, potrebbe essere opportuno condurre test volti a verificare la soddisfazione degli utenti nell'utilizzare l'applicazione. Questi test permetteranno di ottenere informazioni utili per valutare l'usabilità del sistema, il grado di soddisfazione degli utenti e, di conseguenza, effettuare eventuali miglioramenti.

In conclusione, il sistema di monitoraggio e verifica delle procedure industriali presentato in questo lavoro rappresenta un caso d'uso reale di un algoritmo di HOI detection in grado di prevenire e ridurre i rischi associati alle attività lavorative, e di semplificare il processo di formazione di nuovi operatori tramite l'utilizzo di feedback audio e video.

## 5 Bibliografia

- [1] Raied Mehtab Aatish Sharmasanjay mohan Sharma. Augmented reality -an important aspect of industry 4.0. 2021.
- [2] Sven Bambach, Stefan Lee, David J. Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *International Conference on Computer Vision*, pages 1949–1957, 2015.
- [3] Erwin Rauch Benedikt G. Mark, Dominik Matt. Industrial assistance systems to enhance human–machine interaction and operator’s capabilities in assembly. 2021.
- [4] Yixin Chen, Siyuan Huang, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense. In *International Conference on Computer Vision*, pages 8648–8657, 2019.
- [5] Qichen Fu, Xingyu Liu, and Kris M. Kitani. Sequential voting with relational box fields for active object detection. In *Conference on Computer Vision and Pattern Recognition*, pages 2374–2383, 2022.
- [6] Davi Noboru Nakano Jorge Muniz Jr. Gestão Produção, Vagner Batista Ribeiro. Knowledge management and industry 4.0: a critical analysis and future agenda. 2023.
- [7] Bahareh Ghari, Ali Tourani, and Asadollah Shahbahrami. A robust pedestrian detection approach for autonomous vehicles. *arXiv preprint arXiv:2210.10489*, 2022.
- [8] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *Conference on Computer Vision and Pattern Recognition*, pages 8359–8367, 2018.

- [9] Leyla Khaleghi, Alireza Sepas-Moghaddam, Joshua Marshall, and Ali Etemad. Multi-view video-based 3d hand pose estimation. *IEEE Transactions on Artificial Intelligence*, 2022.
- [10] In Su Kim, Hong Seok Choi, Kwang Moo Yi, Jin Young Choi, and Seong G Kong. Intelligent visual surveillance—a survey. *International Journal of Control, Automation and Systems*, 8:926–939, 2010.
- [11] Okan Köpüklü, Ahmet Gunduz, Neslihan Kose, and Gerhard Rigoll. Real-time hand gesture detection and classification using convolutional neural networks. In *2019 14th IEEE international conference on automatic face & gesture recognition (FG 2019)*, pages 1–8. IEEE, 2019.
- [12] Eugene Byrne Zachary Chavis Antonino Furnari Rohit Girdhar Jackson Hamburger Hao Jiang Miao Liu Xingyu Liu Miguel Martin Tushar Nagarajan Ilija Radosavovic Santhosh Kumar Ramakrishnan Fiona Ryan Jayant Sharma Michael Wray Mengmeng Xu Eric Zhongcong Xu Chen Zhao Siddhant Bansal Dhruv Batra Vincent Cartillier Sean Crane Tien Do Morrie Doulaty Akshay Erapalli Christoph Feichtenhofer Adriano Fragomeni Qichen Fu Abrham Gebrselasie Cristina Gonzalez James Hillis Xuhua Huang Yifei Huang Wenqi Jia Leslie Khoo Jachym Kolar Satwik Kottur Anurag Kumar Federico Landini Chao Li Yanghao Li Zhenqiang Li Karttikeya Mangalam Raghava Modhugu Jonathan Munro Tullie Murrell Takumi Nishiyasu Will Price Paola Ruiz Puentes Merey Ramazanova Leda Sari Kiran Somasundaram Audrey Southerland Yusuke Sugano Ruijie Tao Minh Vo Yuchen Wang Xindi Wu Takuma Yagi Ziwei Zhao Yunyi Zhu Pablo Arbelaez David Crandall Dima Damen Giovanni Maria Farinella Christian Fuegen Bernard Ghanem Vamsi Krishna Ithapu C. V. Jawahar Hanbyul Joo Kris Kitani Haizhou Li Richard Newcombe Aude Oliva Hyun Soo Park James M. Rehg Yoichi Sato Jianbo Shi Mike Zheng Shou Antonio Torralba Lorenzo Torresani Mingfei Yan Jitendra Malik Kristen Grauman, Andrew Westbury. Ego4d: Around the world in 3,000 hours of egocentric video. 2022.
- [13] Rosario Leonardi, Francesco Ragusa, Antonino Furnari, and Giovanni Maria Farinella. Egocentric human-object interaction detection exploiting synthetic

- data. In *Image Analysis and Processing–ICIAP 2022: 21st International Conference, Lecce, Italy, May 23–27, 2022, Proceedings, Part II*, pages 237–248. Springer, 2022.
- [14] Jianyong Li, Chengbei Li, Jihui Han, Yuefeng Shi, Guibin Bian, and Shuai Zhou. Robust hand gesture recognition using hog-9ulbp features and svm model. *Electronics*, 11(7):988, 2022.
- [15] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, and Jiashi Feng. Ppdm: Parallel point detection and matching for real-time human-object interaction detection. In *Conference on Computer Vision and Pattern Recognition*, pages 479–487, 2020.
- [16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016.
- [17] Yao Lu and Walterio W. Mayol-Cuevas. Understanding egocentric hand-object interactions from hand pose estimation, 2021.
- [18] Ana I Maqueda, Carlos R del Blanco, Fernando Jaureguizar, and Narciso García. Human-action recognition module for the new generation of augmented reality applications. In *2015 International Symposium on Consumer Electronics (ISCE)*, pages 1–2. IEEE, 2015.
- [19] Francesco Ragusa, Antonino Furnari, and Giovanni Maria Farinella. Meccano: A multimodal egocentric dataset for humans behavior understanding in the industrial-like domain, 2022.
- [20] Francesco Ragusa, Antonino Furnari, Salvatore Livatino, and Giovanni Maria Farinella. The meccano dataset: Understanding human-object interactions from egocentric videos in an industrial-like domain. In *Winter Conference on Applications of Computer Vision*, pages 1569–1578, 2021.

- [21] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28, 2015.
- [23] Evgenii Safronov, Nicola Piga, Michele Colledanchise, and Lorenzo Natale. Active perception for ambiguous objects classification. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4437–4444. IEEE, 2021.
- [24] Bilge Soran, Ali Farhadi, and Linda Shapiro. Generating notifications for missing actions: Don’t forget to turn the lights off! In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4669–4677, 2015.
- [25] Saba Sultan, Ali Javed, Aun Irtaza, Hassan Dawood, Hussain Dawood, and Ali Kashif Bashir. A hybrid egocentric video summarization method to improve the healthcare for alzheimer patients. *Journal of Ambient Intelligence and Humanized Computing*, 10:4197–4206, 2019.
- [26] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4116–4125, 2020.