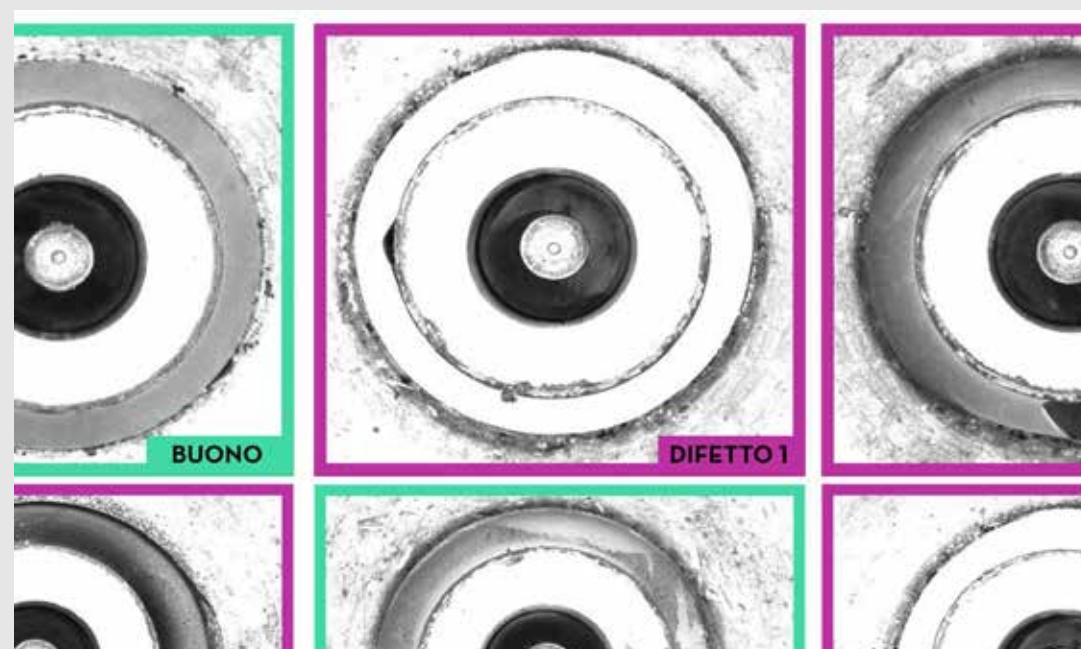


AI-GO

Piattaforma d'intelligenza artificiale no-code per l'industria

- Mette a disposizione un'ampia gamma di modelli pre-allenati
- Raggiunge ottime performance anche con pochi esempi
- Bastano pochi minuti per andare in produzione
- È utilizzabile anche senza competenze specifiche

AI-go RISOLVE PROBLEMI INDUSTRIALI COMPLESSI



CLASSIFICAZIONE

APPLICAZIONI

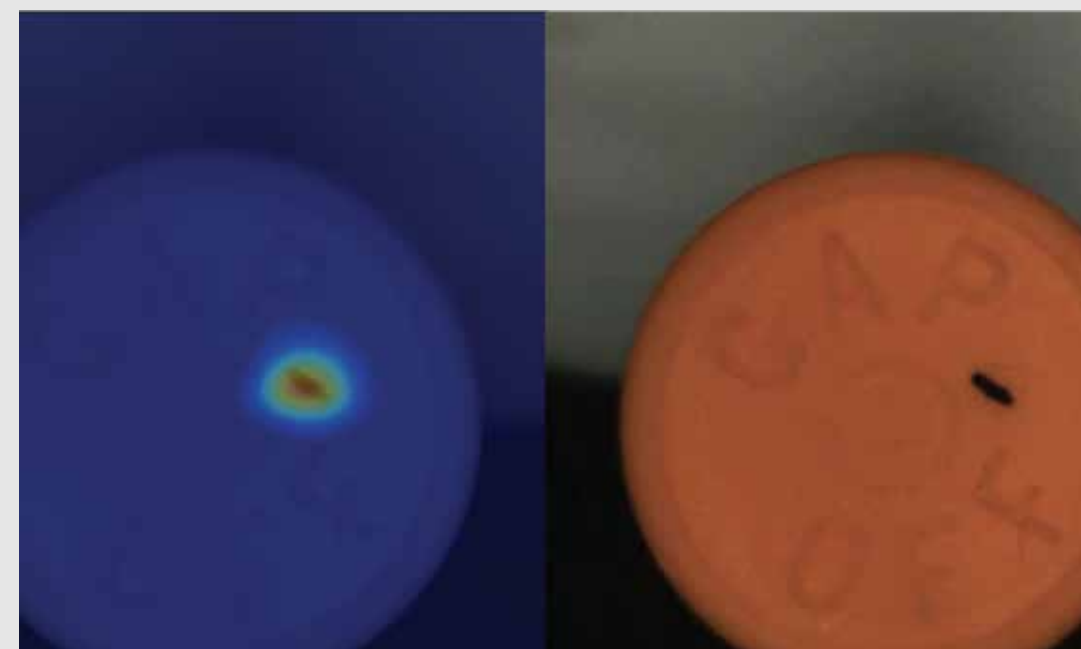
Cernita di pezzi difettosi o anomali, verifica della presenza di parti e del corretto assemblaggio, rilevamento di oggetti.

TASK

Predizione della classe di appartenenza, con restituzione di un valore di confidenza.

AI-go supporta due tipologie di modelli:

- classificazione dell'intera immagine: binaria (buono / scarto) e multiclasse (difetto di tipo 1 / 2 / n)
- classificazione a patch (suddividendo l'immagine in parti più piccole).



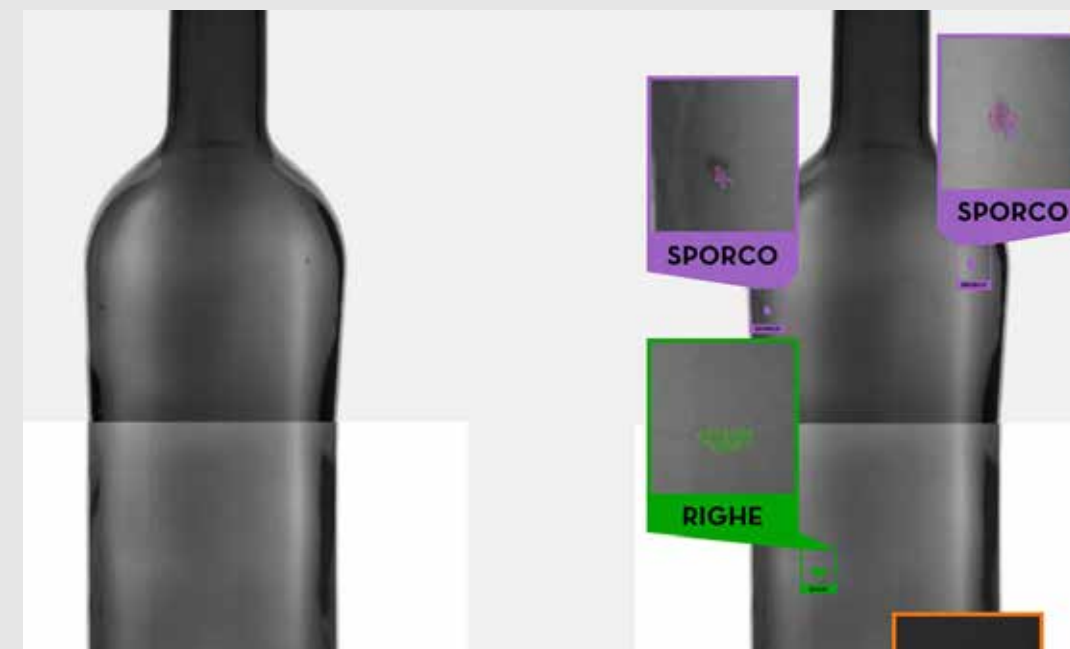
ANOMALY DETECTION

APPLICAZIONI

Identificazione e localizzazione di difetti o anomalie.

TASK

Rilevazione puntuale di un difetto e restituzione di una heatmap che permette all'operatore di visionare l'area "anomala" individuata dall'algoritmo, rendendo comprensibile il giudizio della rete neurale. AI-go impara a rilevare i difetti basandosi esclusivamente su immagini di pezzi buoni (unsupervised learning), senza la necessità di esempi di pezzi scarto.



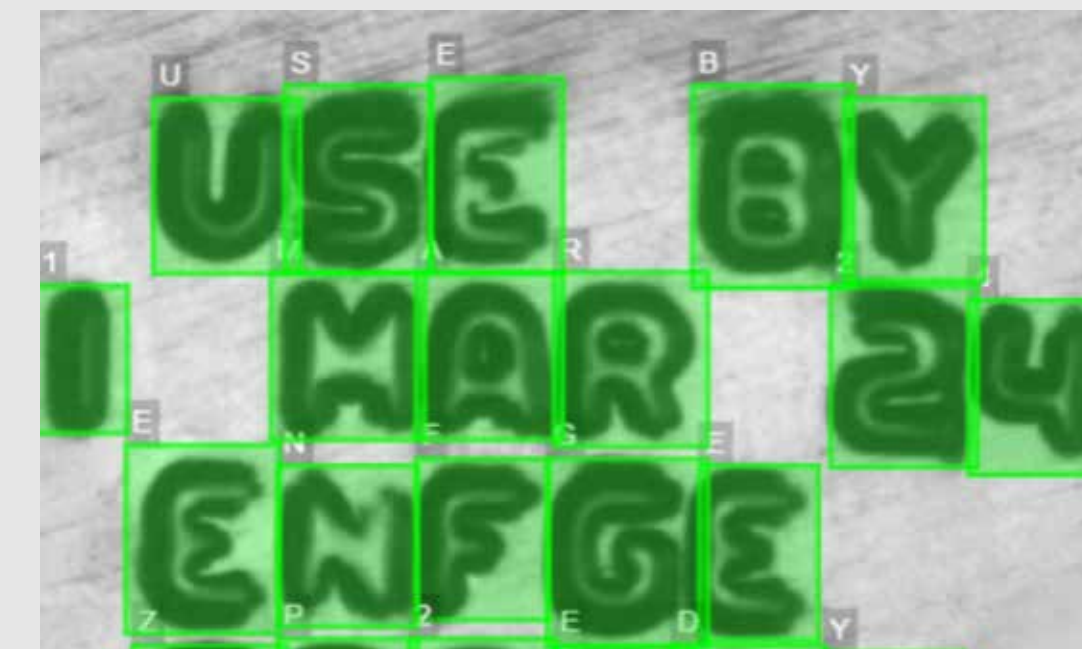
SEGMENTAZIONE

APPLICAZIONI

Cernita e differenziazione dei difetti o di componenti e prodotti, analisi delle geometrie, misurazione dei difetti, ecc.

TASK

Localizzazione puntuale di un difetto e restituzione di una maschera che ne definisce l'area a livello di pixel, consultabile dall'operatore di linea per approfondire e validare il risultato del modello. AI-go supporta modelli di segmentazione multiclasse che permettono la distinzione dei difetti in base alle categorie definite a priori dall'operatore.



OCR - riconoscimento ottico dei caratteri

APPLICAZIONI

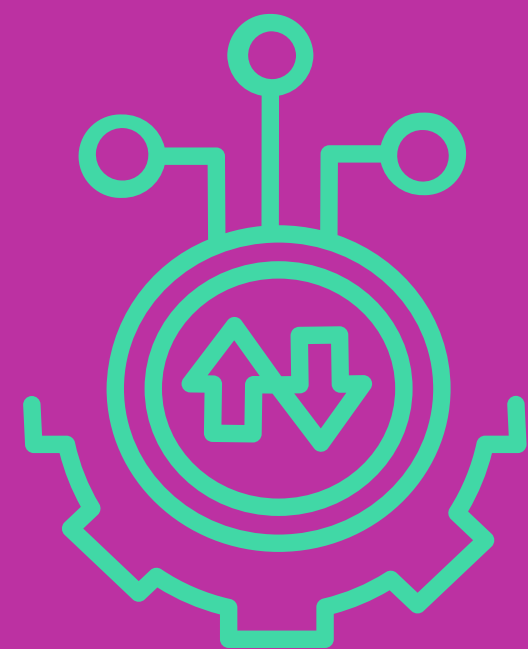
Lettura di etichette e codici, soprattutto quando è richiesta una performance molto elevata.

TASK

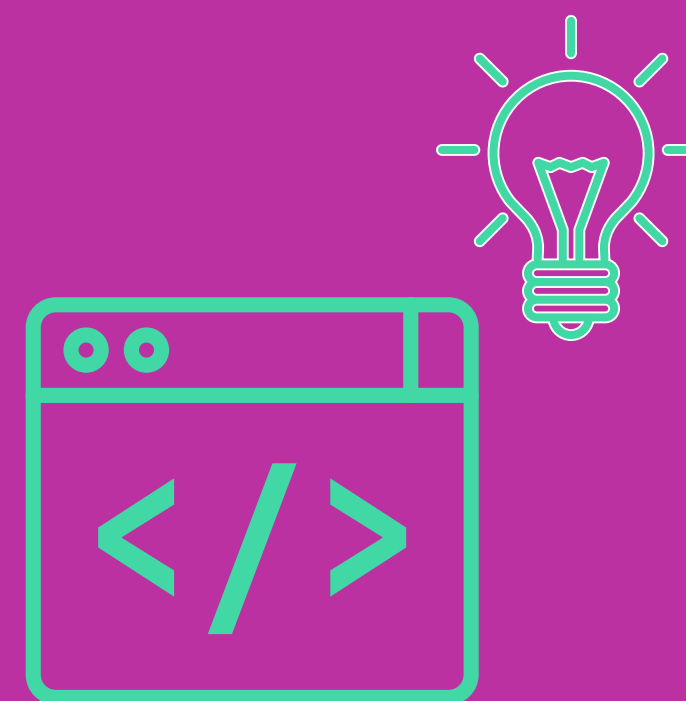
Riconoscimento dei caratteri stampati anche in caso di superfici irregolari (ad esempio fiale, flaconi, bottiglie, barattoli, sacchetti, blister e tubi), di qualità di stampa non omogenea o di scritte in rilievo e con caratteri puntinati e industriali.

5 **PASSI** per integrare AI-go nelle tue applicazioni di visione

PARTECIPA AL PROGRAMMA!

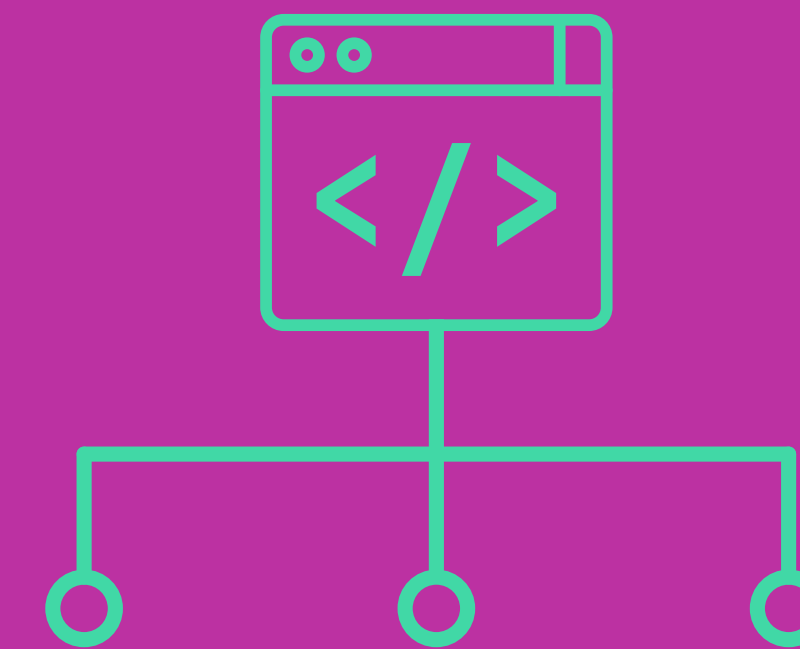


**Facile
integrazione**



**Funzioni
intuitive**

```
aigo_attach_runtime  
aigo_load_model_request  
aigo_image_from_memory  
aigo_predict  
aigo_get_outcome
```



**La stessa
struttura per tutti
i modelli**
(classificazione, anomaly
detection o segmentazione)

5 PASSI per integrare AI-go nelle tue applicazioni di visione

Funzioni Intuitive  

```
// Setup connection AI-go properties
aigo::AigoRuntimeConfiguration runtime_config;
runtime_config.master_url = "169.254.188.190"; //or "localhost";
runtime_config.port = 6379;

// Connect to AI-go runtime!
auto attach_runtime_result = aigo_attach_runtime(context, &runtime_config);
```

1_SETUP CONNECTION

```
// Upload your AI-go model from a given list
aigo_load_model_request(context, "Anomaly_Model_Test_1", runtime_id);
```

2_CARICA IL MODELLO

```
// Upload your image!
cv::Mat img = cv::imread("my_image.bmp");
aigo::AigoImage image = aigo_image_from_memory(img.data() 3 * img.rows() * img.cols());
```

3_CARICA LE IMMAGINI

```
// Build input
aigo::AigoInputData input;
input.image = img;
input.model = "Anomaly_Model_Test_1";

//Performe an inference!
auto id_result = aigo_predict(context, input.get(), 1, runtime_id);
```

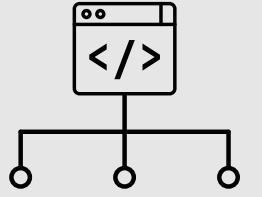
4_INFERENZA

```
//Get outcome
auto output_result = aigo_get_outcome(context, id_result.execution_id, 5000, runtime_id);

//////////
// Serialize output image
auto segmap = output_result.output_datas[0].segmentation;
```

5_OTTIENI IL RISULTATO

La stessa struttura per ogni modello
(classificazione, anomaly detection o segmentazione)

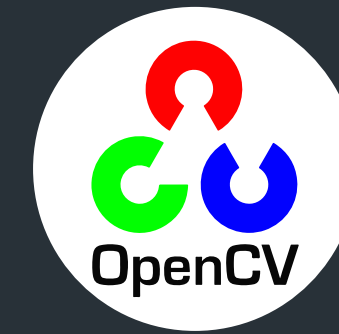
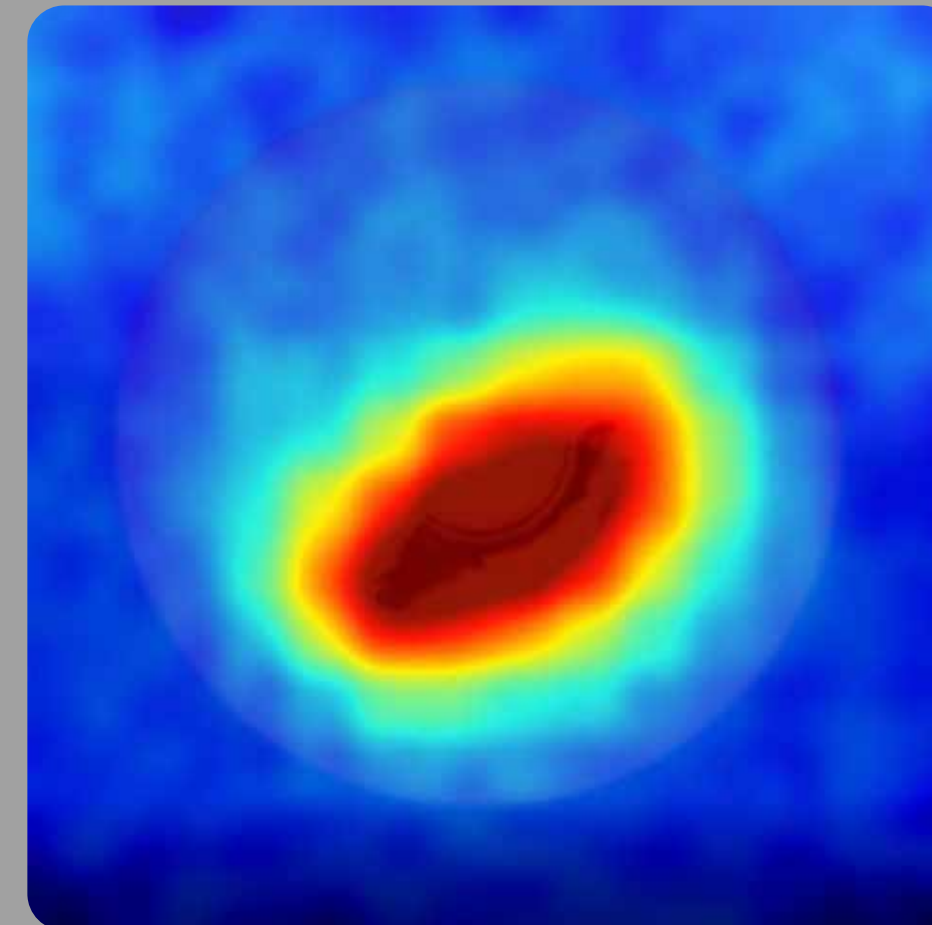


```
// ANOMALY
heatmap = aigo_get_heatmap(segmap);

// Decode the heatmap
cv::Mat imgHeatmapGray = cv::Mat(_segmap->height, _segmap->width, CV_64FC1, heatmap.pointer);

// Convert to 1 channel image
imgHeatmapGray.convertTo(imgHeatmapGray, CV_8UC1, 255.0, 0.0);

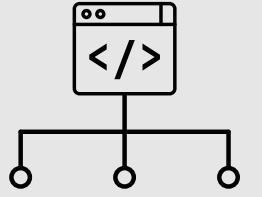
// Blend with original image
cv::applyColorMap(imgHeatmapGray, imgResult, cv::COLORMAP_JET);
double alpha = 0.85;
cv::addWeighted(imgResult, alpha, imgOriginal, (1.0 - alpha), 0.0, imgResult);
```

**INPUT****OUTPUT**

ANOMALY DETECTION

- 1
Ottieni il risultato
con OpenCV
(o una libreria simile)
- 2
Sovrapponi il risultato
all'immagine originale,
ottenendo una **heatmap**

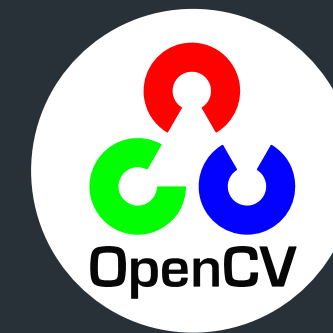
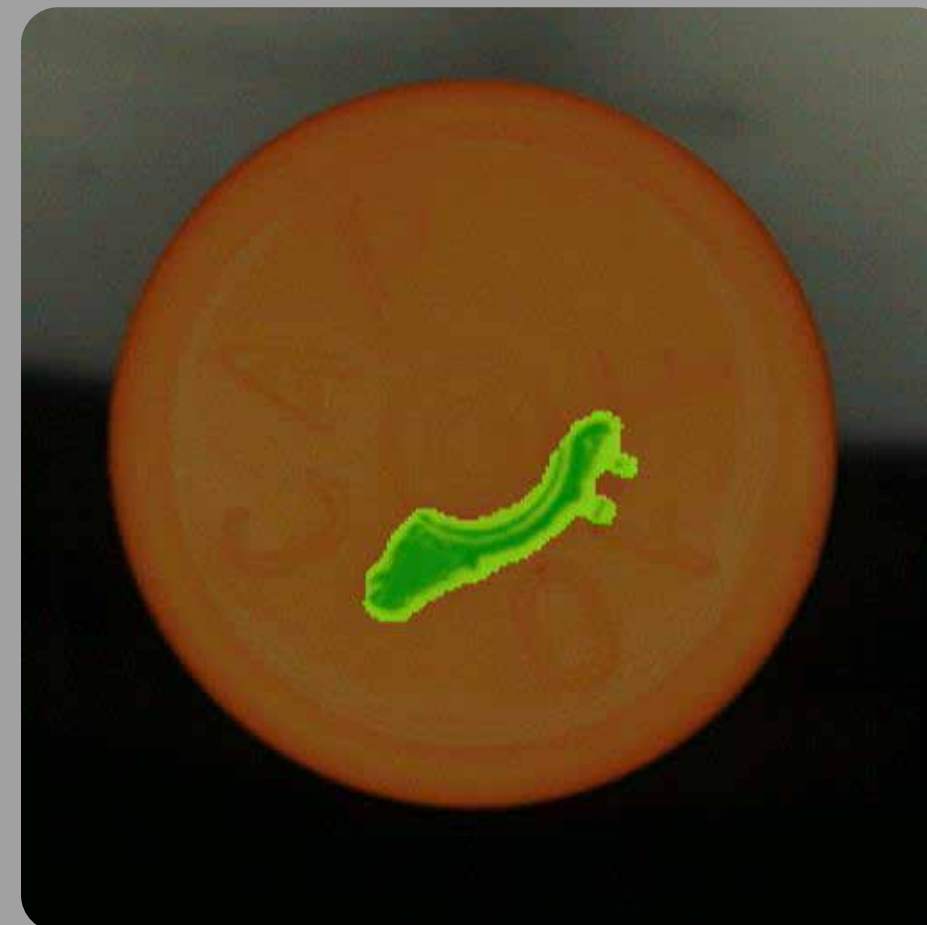
La stessa struttura per ogni modello
(classificazione, anomaly detection o segmentazione)



```
// SEGMENTATION
// Decode the binarymap
auto binarymap = aigo_get_binarymap(segmap);
imgBinMap = cv::Mat(segmap->height, segmap->width, CV_8UC1, binarymap.pointer);

// Make binarymap RGB and blend with original _image
std::vector<cv::Mat> channels(3);
channels[0] = cv::Mat::zeros(imgBinMap.rows, imgBinMap.cols, CV_8UC1);
channels[1] = imgBinMap;
channels[2] = cv::Mat::zeros(imgBinMap.rows, imgBinMap.cols, CV_8UC1);
cv::merge(channels, imgResult);

double alpha = 0.5;
cv::addWeighted(imgResult, alpha, imgOriginal, (1.0 - alpha), 0.0, imgResult);
```

**INPUT****OUTPUT**

SEGMENTAZIONE

- 1
Ottieni il risultato
con OpenCV
(o una libreria simile)
- 2
Sovrapponi la mappa
all'immagine originale,
**evidenziando l'area
segmentata**

ORÒBIX

Machine Vision
GeeX Program

CONTATTACI!

Orobix srl
via Gabriele Camozzi 144
24121 Bergamo - Italy
+ 39 035 017 0561

www.orobix.com
info@orobix.com

