

# STAGE LINDA LUDOVISI

Preprocessing the images with OpenCV

# 1 - WHICH IMAGES ?

- The images (frames) were extracted from several videos which show some food products on the conveyor belt.
- Each video (**.h264**) was made of 3500-4000 frames.
- Each frame was then preprocessed.

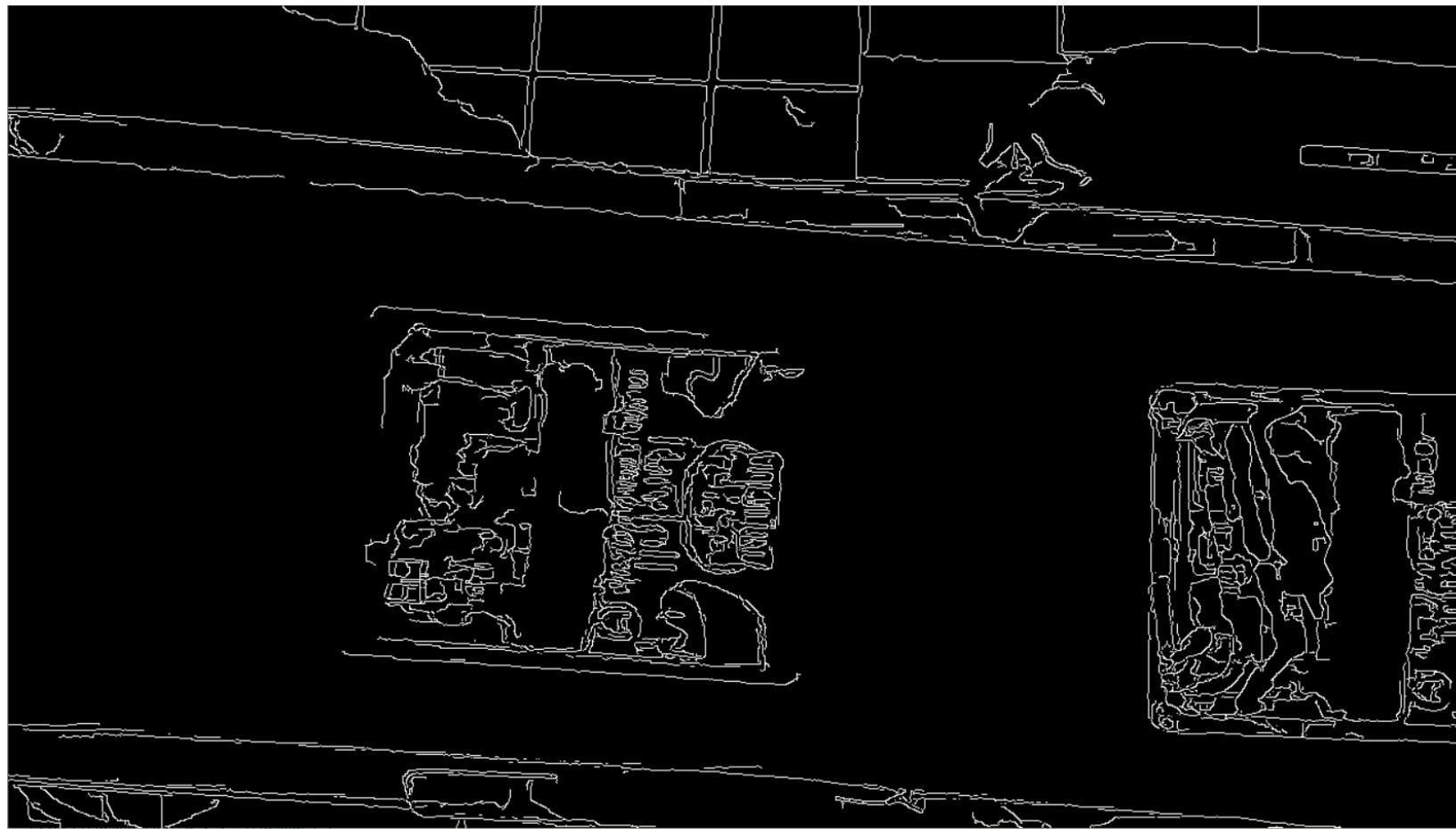


# 2 - HOUGH LINE TRANSFORM

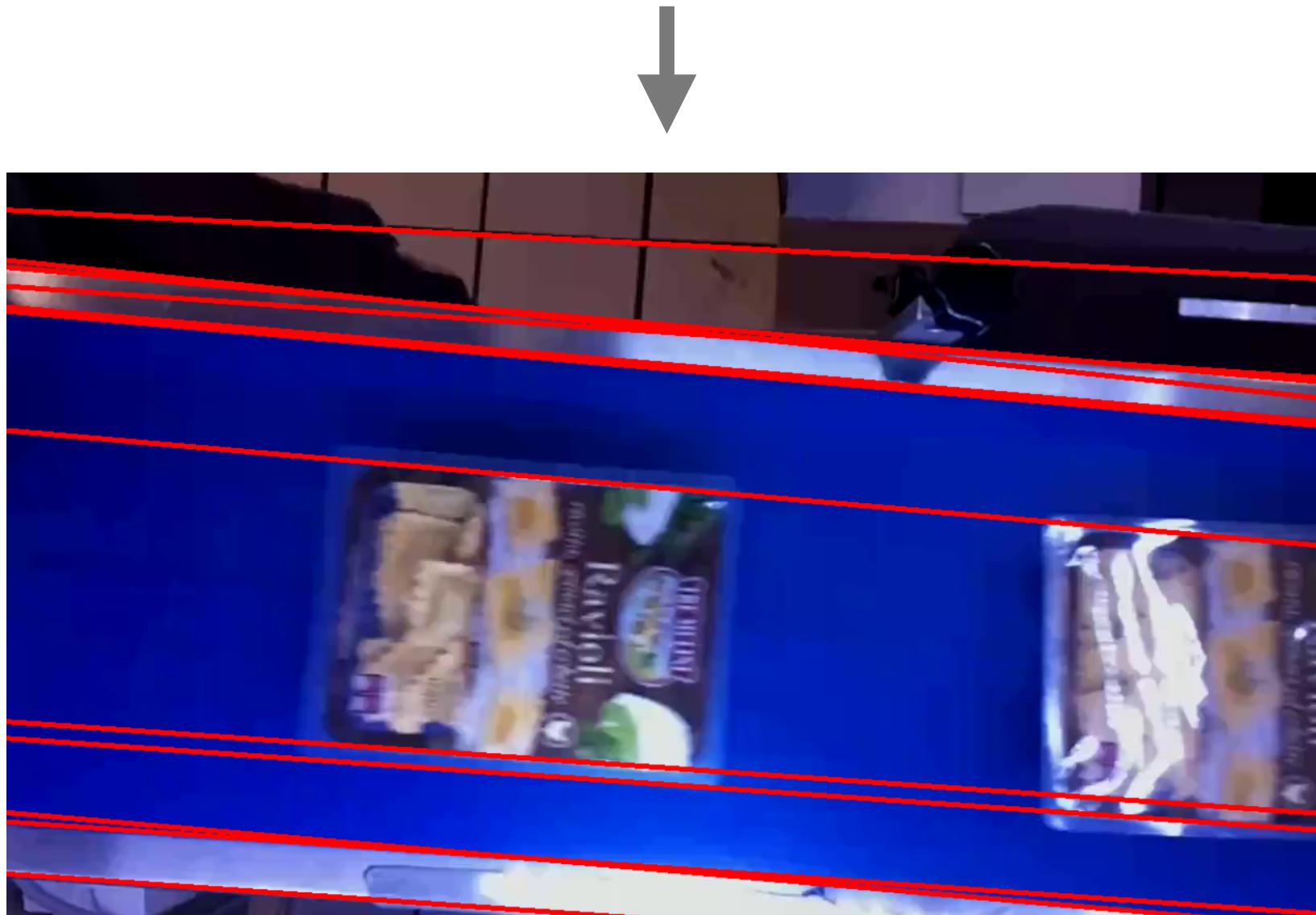
- The first step of preprocessing consisted in performing the **Hough Line Transform**.

*All the code was written in Python, OpenCV library was used.*

- To perform it, a generic frame was selected and after applying **Canny Detector** on the image to show the edges, the Hough Lines were drawn down.

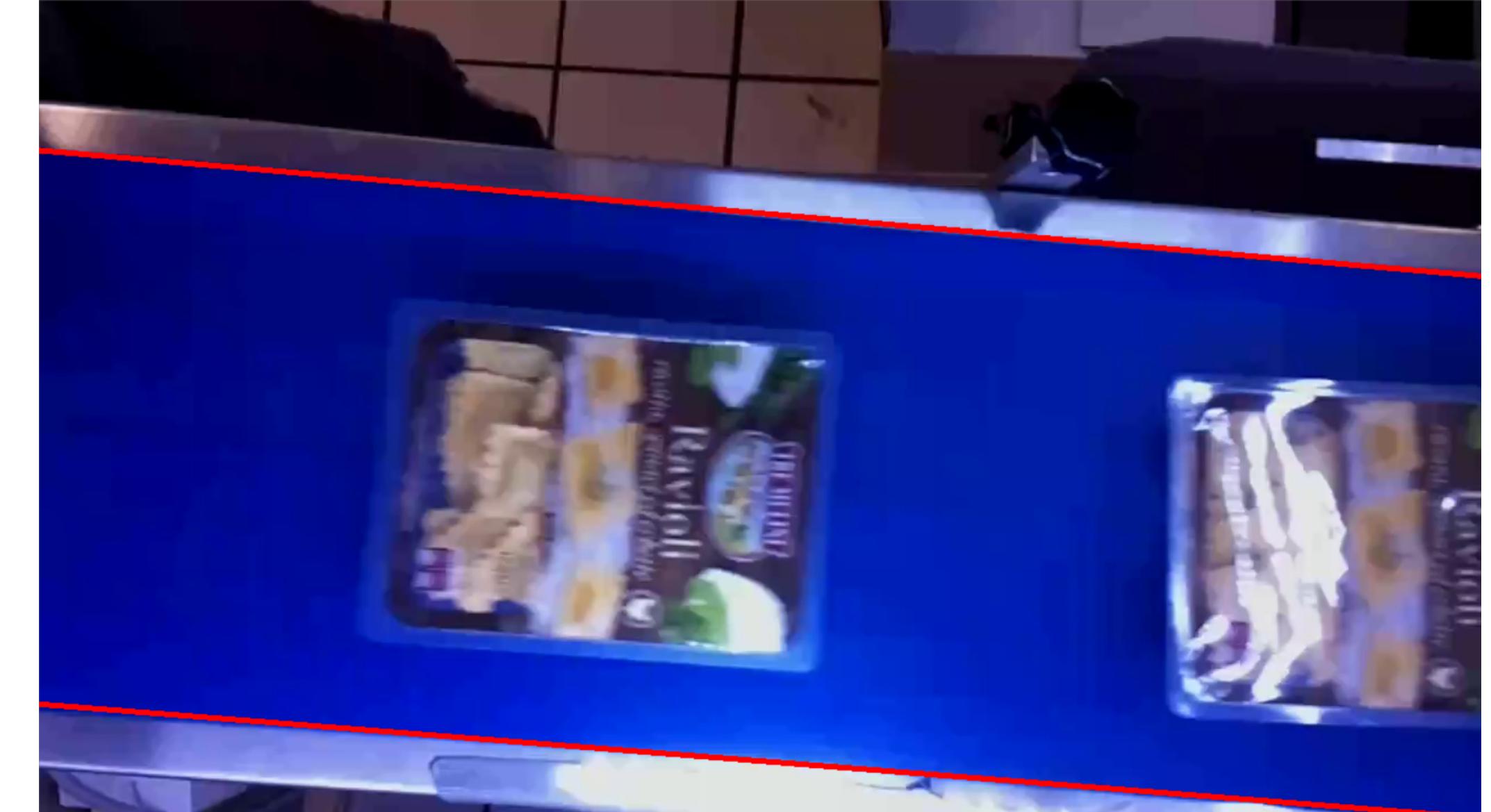
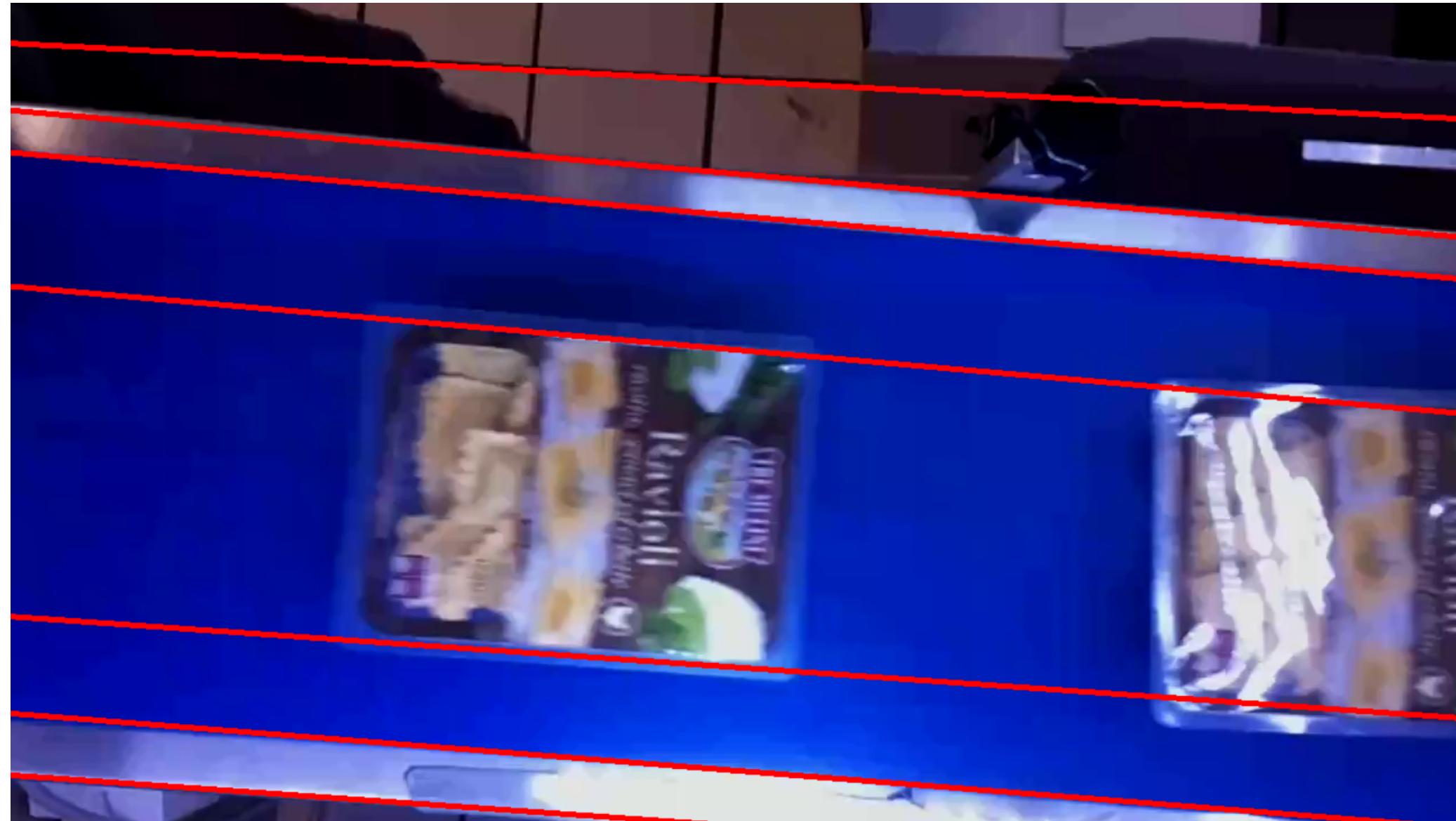


Canny  
Detector  
Output



Hough  
Lines  
Output

### 3 - CLUSTERIZATION TO REDUCE NUMBER OF LINES

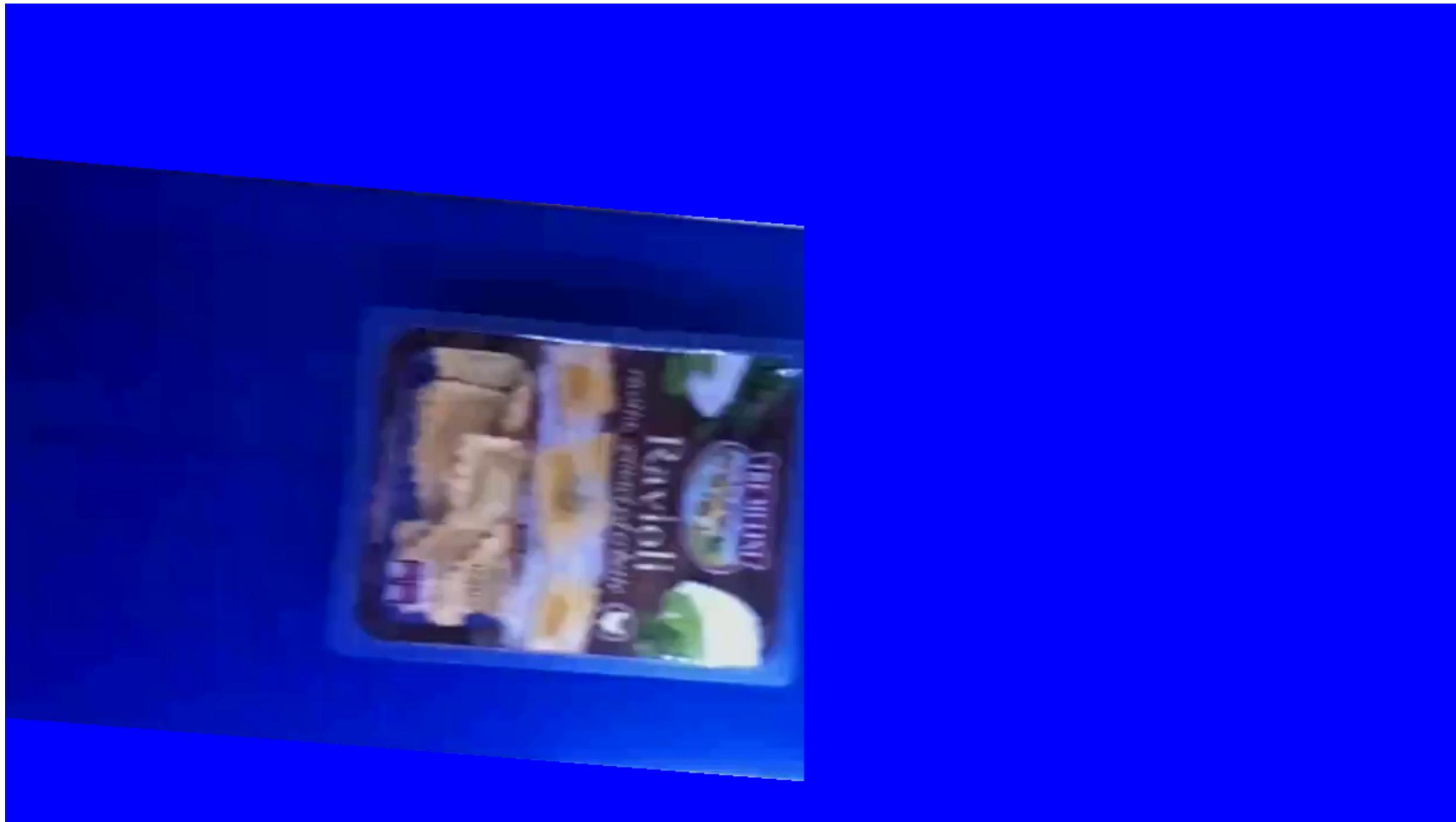


A **KMeans**( $k=7$ ) was performed over values of  $\rho$ ,  $\theta$  to obtain the main, non-overlapped lines (clusters).

Finally, the two main lines (the ones referring to the conveyor belt) were selected.

## 4 - MAKE PIXELS BLUE

After having detected the **two main lines** (they are the same for ALL the frames), the following edit was done to the images:

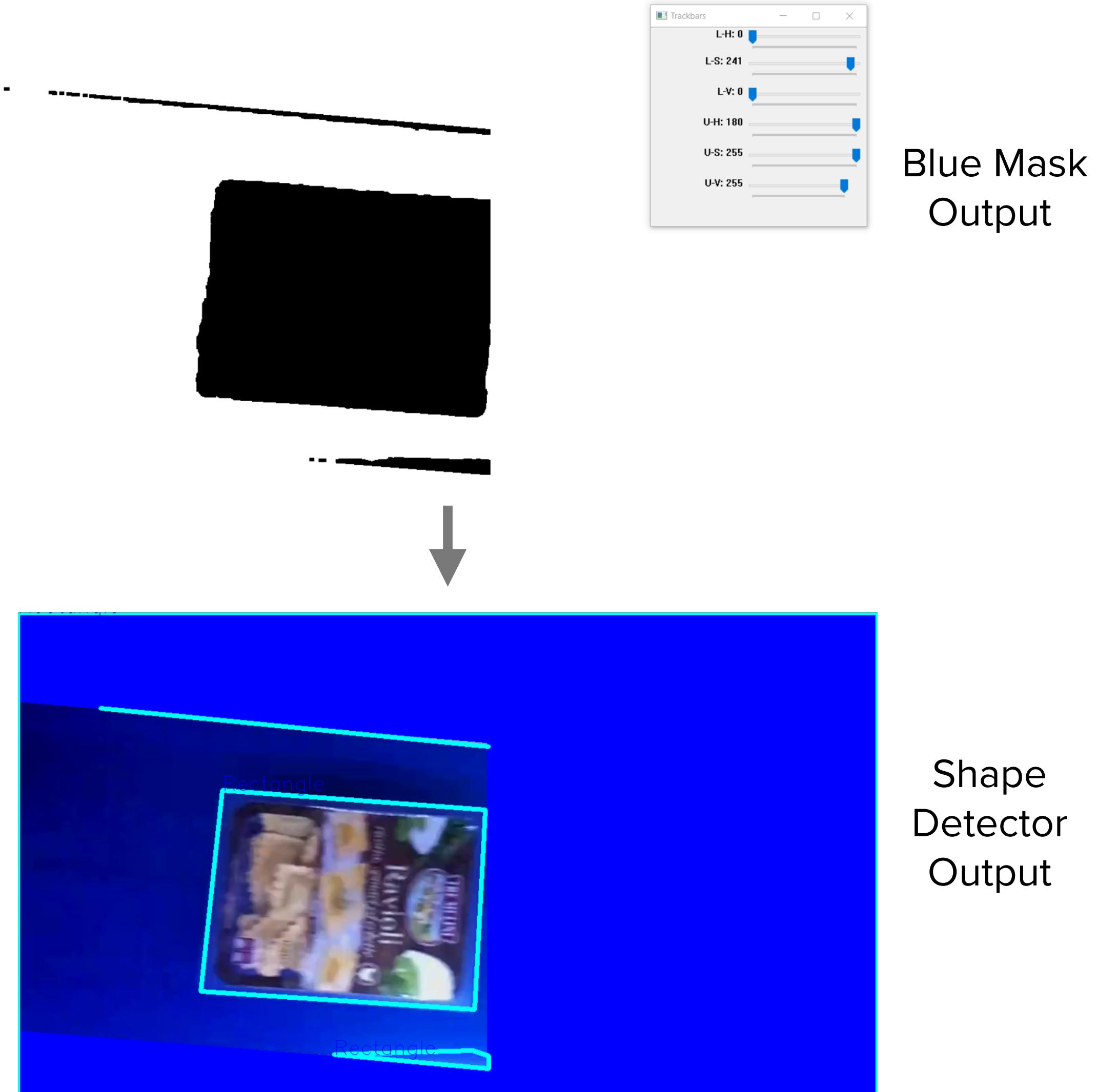


I also decided to make **half the image blue**: this decision was due to the presence of neon lights which produce reflections on the products.

**All the images are preprocessed in this way** (also using the same coordinates for the lines).

# 5 - SHAPE DETECTOR

- Then, for each frame a mask was created: it was made in a way that only the blue pixels were detected. This was performed by selecting the correct values of blue in **HSV color space**. This generated a binary image on which the shape detector phase was performed.
- To perform shape detection, the contours were found and then approximated with **Ramer-Douglas-Pecker algorithm**.



## 6 - FINAL IMAGE

Finally, if the rectangle (with an appropriate area) was found, it was cropped from the frame and the following image was created:

