* The steps to using the script on new footage:
  + ROI is now mandatory to provide perspective correction. Four points need to be found using any software which allows for spying the coordinated. ImageJ works well.
  + True width and height of the ROI must be supplied for perspective correction to work well and generate 1:1 aspect ratio corrected image.
    - If true width and height in mm is not known, provide estimates which will allow the script to give the corrected ROI correct 1:1 ratio.
    - Pixel-to-mm calibration is applied based on the ROI width at the moment.
  + Run the script with live preview to check that the perspective correction works and leeks are being picked up.
  + If the script is picking up bunched up leeks, or missing small leeks then maximum and minimum detection area thresholds need to be tweaked. Areas of detected leeks are saved together with width measurement, so they can be used to spot outliers and set better thresholds.
  + The position of measurement line from the white tip can be tweaked.
* List of decisions script makes when looking for valid leeks to measure:
  + Area of small bounding box between minimum and maximum thresholds
  + Aspect ratio of the small bounding box between minimum and maximum
  + Orthogonal bounding box not contacting the image edges
* The focus card is much too small to help with perspective correction. I had to come up with my own way to calibrate the image, including perspective correction. Here are the steps I took:
  + Notice that the 12-inch ruler is approximately the length of 10 eyelet columns.
  + Assume that the eyelets are arranged in a uniform manner.
  + Use eyelets on a still frame to align four-point ROI. The ROI I chose was 19 eyelet columns wide and 13.5 eyelet rows tall.
  + Based on the above assumptions of uniform eyelet spacing calculate the true height and width of the ROI in mm.
  + Use the ROI to perform 4-point perspective correction (openCV function).
  + Calculate pixel-to-mm factor based on the dimensions of the new frame and the true dimensions.
    - After 4-point perspective the pixels should have 1:1 ratio.
  + When I measured the dimensions of the focus card using my 'calibration', the dimensions were off by about 3% in the direction of the belt travel (where most of perspective distortion resides) and 1% across belt. The error on the leeks should lie somewhere between those two values.
  + Note: if we keep using the conveyor with eyelets (and it's a good choice since it's sheltered and the leeks are cleaned) ask Tom to take some dimensions of the eyelet grid layout and we can use that as a quite good measuring stick.
  + Future work: we could get a more accurate spherical lens correction using eyelets of the conveyor as reference points.
* There is something very technically odd about the newest video (the one 17 minutes long). OpenCV hated it and I had to transcode it.
* Because there was so much diversity in the footage we received (type of belt, light conditions, direction of travel, lens distortion, apparent size of leeks) I really cannot provide a one-size-fits-all script. It will require tweaking until the videos you get become more standard. I've collected all the tweak-able variables at the start of the script along with comments on what they do. You should have full control without delving into the code.
  + There are so many options and variables that ideally the script would come with a GUI, that I didn't have time for. I think that second-best thing is to just open the script and explore the variables I collected at the start.
  + Running the script with the display function active shows 3 windows: full colour perspective-corrected ROI with boxes around recognised leeks, the mask generated by background model or threshold function and last- the cropped leek with overlaid measurement. This output will help in tweaking the script variables.
* The porous belt footage is much 'cleaner' than the footage from the other conveyor. It responds well to Otsu thresholding, without using the background subtraction model. It helps that the conveyor is green, since the contrast in red channel is very high. I will leave the background subtraction model in as a toggle, but it complicates and slows the script, so should be used only when needed.
* I included a nice progress bar function, but if activated (via toggle variable) it will require a tqdm package to work.
* Initially I thought of ROI as optional, but now I think it should be mandatory. It serves triple function: preventing bad detections outside of conveyer, greatly speeding up the computation and crucially- perspective correction.