Inspection

Problem statement

Fermentic Inc produces tips through injection molding with quite high rejection rate. To make things worse, the inspections are performed by manual labor; both time consuming and error prone.

There have been cases of false positives; tips that have passed QC but actually not suitable for usage. And the management is greatly concerned about the possibility of false negatives; good tips that got rejected due to inappropriate judgment.

To improve yield, FI is considering an automation solution and vision-based inspection is deemed as a low-cost solution for the matter.
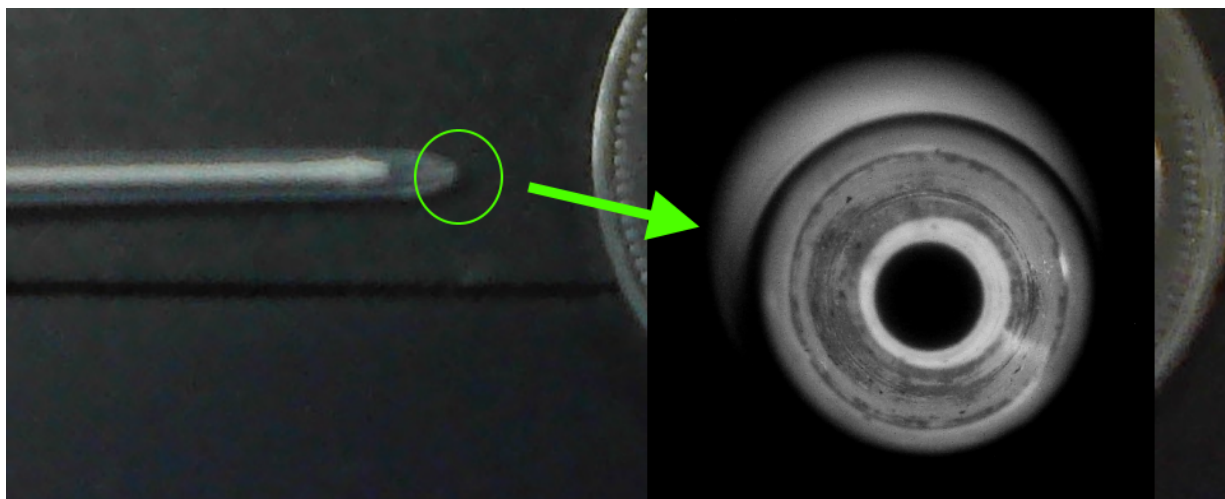
Some vendors have been approached, but although all boasted how great their solutions are, none provided a real (and plug-n-play) solution for this particular case. In the end it was decided to develop the solution in-house.

Requirements:
- Reject obvious bad tips
- The pipeline can't be too rigid; otherwise the yield would be very low.
- There are many aspects to verify, but for now the focus is to evaluate the end of the tip.
- Provide overlay of detection results to allow pipeline verification.
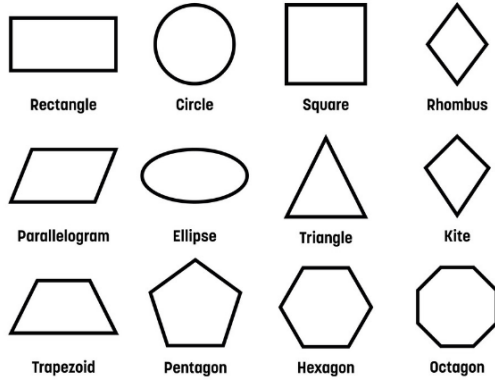

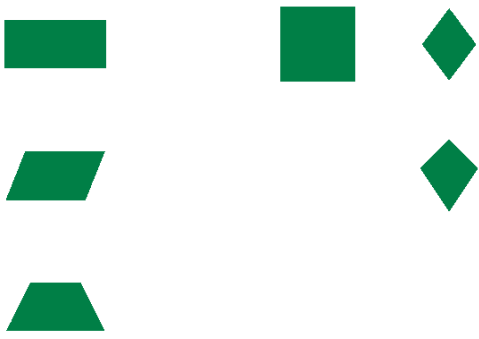The end of the tip (compared to Rp 1000 coin)

The end of the tip - bottom view with magnification

Terms

Term	Meaning
Tip	Disposable component to aspirate and dispense liquid
False positive	Qualifying bad tips as end product
False negative	Rejecting tips that should've been working well
Expected precision / accuracy	Not determined yet, the performance[1] "promised" to the customers. Example: 10mL accuracy with +/- 1 mL precision tolerance. Final figures to be defined based on result analysis.
OK / NG	OK / NO GO. The former identifies a tip as a good one, the latter should be thrown away.

---

[1] The management might want to decide to sell the tips into 2 variants: the premium (with higher specifications) and the economic variant for those deemed defective but still usable within tolerance.

Annotation overlay		
	A sample input image	Overlay for quadrilateral detection pipeline.

Expected output

1. Please ask questions if you don't understand the problem, we are happy to answer.
2. Describe the problems inferred from looking at some sample images.
3. Propose solution(s) to the problem found above, describe it in short paragraph/diagram/pseudo.
4. Show us the code :) provide working proof of concept
   a. Provide the solution in a git repository or zipped directory.
   b. We expect to just clone the repo (if it is in zip file, just extract) and run, if the work requires 3rd party components, provide instrumentation (scripts/cmake/etc) to automate this.
   c. Use the input images on the attached zip folder on the email (**Test Set - Vision Test**). Please notice that not all input images can be processed, some images are considered as NO GO (corrupted). Your image processing pipeline should be able to detect such conditions. Please see the 'terms' table above.
5. A complete solution would be nice, but partial work will be considered, i.e. detecting tip circle, measuring inner and outer concentricity, detecting bad tip (not a clean circle).