Automatic Detection of Defective Flask Labels

Dmitry Chetverikov[3,1] and Ákos Zarándy[1,2]

[1] SZTAKI, Hungary
csetverikov@sztaki.hu
[2] Pázmány Péter Catholic University, Hungary
[3] Eötvös Loránd University, Hungary

Abstract. In the framework of an industrial project, we address the problem of label inspection on roughly flat, plastic bottles (flasks) containing dishwashing liquids, shower gels, skin care cremes, etc. A flask is placed vertically in a special box, rotated around its vertical axis, illuminated and viewed by a camera. Several images are acquired when the front label is found to be parallel, or close to parallel, to the image plane of the camera. Defective labels are detected and their flasks removed for re-labelling. Three typical types of label defects are considered. A novel method for automatic label inspection is proposed. Results of initial tests on numerous flasks of different kinds are shown and discussed.

1 Introduction and Previous Work

In this paper, we present a novel label inspection system developed in a recent industrial project[1] whose goal was to automate the process of label defect detection using efficient image acquisition, processing and analysis.

The literature related to research and development for label defect detection is quite limited because of the industrial character of the problem. More information can be found in the related patents, so we are going to discuss some of them, as well, despite the fact that most of the patents do not provide all necessary technical details.

Short paper [2] provides a brief description of a method for checking the presence of a label on the inspected bottle. Potential label defects are not considered. PhD dissertation [6] provides a very initial overview of bottle checking tasks including the detection of missing, empty (not printed), and wrongly oriented labels. Otherwise, the dissertation is devoted to the detection of wrongly filled bottles rather than label checking.

Study [5] presents an experimental setup for checking labels printed on the surfaces of bottles and flasks. A detailed description of the technical and algorithmic components is provided. An image of the printed text is acquired with a high-resolution camera, processed, binarized, then compared to the ideal, reference text in order to detect printing detects. In our study, we address the problem of detecting the defects caused by the tool placing the label on the surface of the bottle rather than the quality of printing. Our task does not require a high-resolution camera since the defects are visible in relatively low resolution, as well.

[1] "Plastic bottle extruding and labeling development for building new innovative, environment friendly packaging materials and technology (GINOP-2.2.1-15-2017-00075)."

Early patents [9] and [3] describe sophisticated mechanical devices equipped by photo-sensing scanners that scan the arriving bottles and check them for the presence and proper alignment of the label. Although the patents contain some interesting ideas, and the proposed systems are claimed to be able to cope with transparent and semi-transparent bottles, the mechanism of the horizontal line-by-line photo-scanning is definitely obsolete, imprecise, and not robust.

A more recent patent [10] proposes an optical inspection system for defect detection in colored labels placed on the side walls of cylindrical beverage cans or other types of bottles. The cans arrive in fixed vertical position but arbitrary orientation around their axes. They are illuminated and viewed by an optical head that scans them and generates two kinds of color signatures. After a learning period on a large number of good labels, two reference signatures are obtained and checked against the inspected labels. A can is rejected if its label significantly deviates from a reference signature.

Finally, patent [8] describes a framework system for detect detection *within* semi-opaque containers such as labeled plastic or glass bottles and flasks. To find a non-conforming object of unknown shape and location within the bottle, two consecutive light sources are used, and two images are acquired. The difference image is then obtained in order to eliminate the label and find the defect.

The structure of the paper is as follows. In Section 2, we specify the task of label inspection and discuss its challenges. The proposed defect detection framework involving a number of distinct algorithmic steps is presented in Section 3. The results of testing the proposed system are shown and discussed in Section 4. Finally, Section 5 concludes the paper by a brief description of open questions and future work.

2 Task and Challenges

Given an input flask image and some reference image data discussed later in Section 3, the goal of the proposed method is to provide statistical data sufficient for reliable classification of the flask label into two classes, good label and defective label. Defective labels are rare, and they cannot be used for efficient learning.

Examples of good labels are shown in Figure 1, while a few typical label defects are illustrated in Figure 2 where the most distinct and easy to detect defect category is a partially missing (part.miss.) label, while 'mispl.' means misplacement. The typical resolution of the images varies in the range of 400×800 to 300×1100 pixels depending on the shape of the inspected bottles.

The detectability of the other two types of defects strongly depends on the degree and kind of misplacement or crinkling. As discussed later, a correct and well-visible but wrongly placed, e.g., skewed, label can pose a problem. Similarly, a crinkle can be very narrow, resulting in a small defect area.

Further potential challenges are as follows:

- **Varying illumination**. Despite the controlled conditions, flask illumination varies within image and in time.
- **Varying orientation and image** of flask due to imprecise fixing and rotation around the vertical axis.

good 1 good 2 good 3 good 4

Fig. 1. Set 1: Good labels.

misplaced 1 part.missing misplaced 2 crinkled

Fig. 2. Set 1: Defective labels.

- Moderate **perspective distortion** of label since the camera is relatively close to bottle.
- Large **featureless areas** in some kinds of labels, which can make the matching of the reference and input labels more difficult.
- **Transparent bottle** with labels on both sides, where the image of the viewed label can vary depending on flask pose.
- **Very fine textures** formed by texts with tiny letters are hard to compare under limited image resolution and positioning precision.

The proposed method attempts to address most of these challenges. However, some of them still pose problems and need further analysis for future development of the method.

3 Method for Defective Label Detection

The proposed method operates with the input flask image and additional input data. For each type of flask with a given label, a reference flask image with defect-free label is manually selected for future comparison to the input labels. Then, the label area is manually extracted from the reference image, which provides the *reference label*. This is done only once for each type of bottle and label. A reference label is a rectangular framework image with the label overlaid on the dark background. Finally, the binary *label mask* is automatically extracted from the reference label image by thresholding and post-processing. The latter includes gap filling and slight shrinking to ensure comparison within the label area. Figure 3 illustrates the additional input data of the proposed method.

reference flask reference label label mask

Fig. 3. Sample reference flask and input of the proposed program, reference label and its mask.

The major steps of the proposed method are as follows:

1. Extract distinct and robust local **SURF features** [1] in the reference and input images.
2. Assign a **SURF descriptor** [1] to each feature.
3. Using the features and their descriptors, find the **perspective mapping** of the reference label onto the input image.
4. **Map the reference label** and its mask onto the input image frame.
5. **Shift-correct intensity** in the reference label and the corresponding (masked) area of the input image by subtracting their mean values.
6. Calculate pixel-wise absolute intensity differences between the reference and the input in the mask area to obtain the **difference image**.
7. Calculate the **histogram and statistics** of the difference image.
8. Apply a preset threshold to the difference image within the mask to obtain the binary **defect mask**.
9. In the defect mask, **remove small connected components**, i.e., those with area below a preset area threshold.
10. Calculate the total **defect area** after the thresholding and output it along with the difference image statistics.

Figure 4 illustrates the process of feature extraction and matching in the reference and input images. Robust feature correspondences are established under the assumption of possible perspective distortion between the reference and its counterpart in the input. It is also assumed that the surface of the flask carrying the label is roughly flat, and most of the label is visible in the input. The perspective mapping allows the method to cope with moderate perspective distortion, including rotations and scaling due to the potential variation of the distance between camera and bottle.

After the feature matching, the reference label and the label mask are mapped onto the input image frame as illustrated in Figure 5. (The input image is shown in Figure 4.) The mapping is robust to wrong correspondences and feature outliers assuming that the number of correct correspondences is sufficient to apply a RANSAC-like procedure [4]. In steps 1–4, we use functions *SurfFeatureDetector*, *SurfDescriptorExtractor*, *FlannBasedMatcher*, *findHomography*, and *warpPerspective* provided by the OpenCV library [7].

After the mapping, we have the reference and the input labels matched, with the reference mask specifying the part of the input image where the comparison of the two labels should be done. However, direct comparison of the labels can be misleading because of the varying illumination mentioned in Section 2 among the challenges of the task. To cope with this effect, we shift-correct the intensities of the two labels within the mask by subtracting the corresponding mean values.

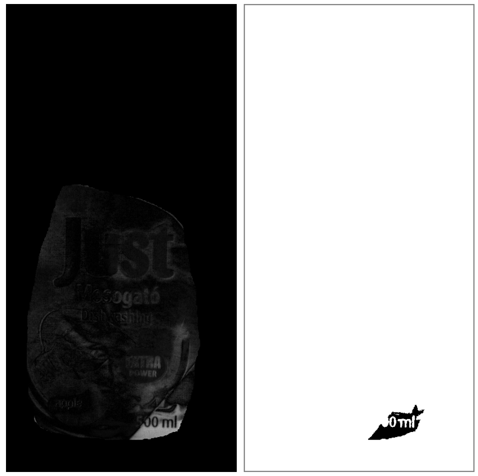
After such correction, the pixel-wise absolute difference between the labels is an adequate representation of the deviation from the reference. A relatively low, conservative preset difference threshold indicates the locations of significant deviation which are potential defects. Figure 6 shows the absolute difference image and its thresholded binary version for the input image of Figure 4.

Finally, we calculate a number of statistical features that can be used to discriminate between correct and defective labels. The most straightforward feature is the number

Fig. 4. Illustration of feature matching.

reference label label mask

Fig. 5. Results of mapping onto input image frame.



absolute difference thresholded

Fig. 6. Absolute difference within mask before and after thresholding and removal of small components.

of the above-threshold pixels in the thresholded difference image, that is, the area of the potential defects. We also calculate the basic statistics of the absolute difference image, namely, the mean and the median values, the standard deviation (variance), and the mean absolute deviation from the mean.

Altogether, we have five features, but their discriminative powers are different. For example, the median tends to discard outliers that are indicative of defects, while the standard deviation (stdev), on the opposite, is sensitive to the outliers. Figure 7 provides examples of absolute difference histograms of a defect-free and a defective label, where the difference in variance between the two is clearly visible. After preliminary tests, we decided to use two indicators of defects, the standard deviation of the difference image and the area of its thresholded version.

The proposed method has two main parameters, the *difference threshold* and the *area threshold*. The former specifies the lowest difference considered potentially defective, the latter the smallest connected component area to be taken into account: smaller components are removed in post-processing. The default values of the two parameters are 80 (greylevels) and 40 (pixels), respectively. These values should be adapted to the types of label and bottle, as well as to the illumination conditions.

The overall *processing time* per image was within 1 sec on a 10-years old PC without any GPU support. This time can be significantly reduced if the system is implemented on a much more efficient, modern computer with GPU support.

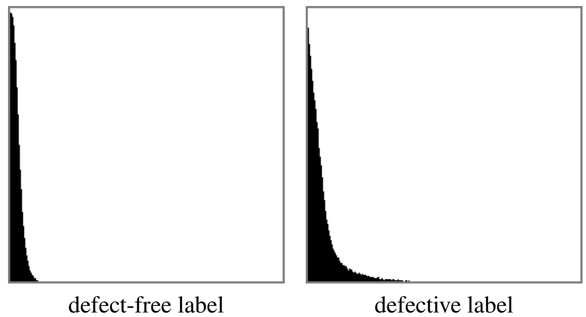


Fig. 7. Sample absolute difference histograms of a defect-free and a defective label.

4 Initial Test Results

In this section, we present numerical results for four different kinds of flasks and labels and demonstrate that the selected features, the defect area the the standard deviation, can be efficiently used to detect defective labels. Each set consists of four good labels (flasks) and four defective labels of three different kinds, misplaced (mispl.), partially missing (part.miss.), and crinkled.

Set 1. The good labels of this set, with the flasks, are shown in Figure 1, while its defective labels are demonstrated in Figure 2. Table 1 contains the feature values for the good labels, Table 2 for the defective ones. One can observe significant differences in both features between the correct and the defective labels. In particular, all noisy connected components, if any, were removed in the good cases, while the overall defective areas in the bad cases are convincingly large. In the standard deviation, the border between the two cases seems to be around 12–15.

feature	good 1	good 2	good 3	good 4
defarea	0	0	0	0
stdev	7.78	8.12	6.92	7.60

Table 1. Feature values for **good** labels of set 1.

feature	misplaced 1	part.missing	misplaced 2	crinkled
defarea	2830	3418	656	21201
stdev	22.56	22.62	15.39	38.80

Table 2. Feature values for **defective** labels of set 1.

Set 2. For simplicity, we do not show the good labels of the remaining sets; their characteristic feature values are only provided. For set 2, the largest standard deviation is 7.95, with all defect areas being zero again. The defective cases of the set are shown in Figure 8, their feature values in Table 3. Here again the differences between the two subsets are convincing enough.

misplaced 1 part.missing misplaced 2 crinkled

Fig. 8. Set 2: Defective labels.

feature	misplaced 1	part.missing	misplaced 2	crinkled
defarea	2663	2827	1185	692
stdev	29.27	25.92	24.18	20.56

Table 3. Feature values for **defective** labels of set 2.

Set 3. The defective cases of the set are shown in Figure 9. The feature values are given in Table 4. For the good part of this set, the largest standard deviation is 10.28, and all defect areas are zero as before. We observe that in this case two of the defect areas and three of the standard deviations are small, which can potentially affect the reliability of decision making. The sources of the problem will be discussed later in Section 4.1.

feature	misplaced 1	part.missing	misplaced 2	crinkled
defarea	153	72	72	3533
stdev	10.71	11.11	10.75	29.25

Table 4. Feature values for **defective** labels of set 3.

Set 4. The defective cases of the last set are demonstrated in Figure 10, the feature values are presented in Table 5. For the good part of this set, the largest standard devia-

misplaced 1 part.missing misplaced 2 crinkled

Fig. 9. Set 3: Defective labels.

tion is 9.05; all defect areas are zero. While the defect areas are large or relatively large, half of the standard deviations are quite small similarly to the previous set. The reasons are discussed below.

misplaced 1 part. missing misplaced 2 crinkled

Fig. 10. Set 4: Defective labels.

4.1 Analysis of problematic cases

The missing part of the second label of Set 3 (Figure 9, part.missing) contains few features and is almost transparent, i.e., does not significantly differ from the background

feature	misplaced 1	part. missing	misplaced 2	crinkled
defarea	9882	269	339	1929
stdev	40.43	12.65	13.51	23.45

Table 5. Feature values for **defective** labels of set 4.

plastic. The wrong poses of the otherwise good misplaced labels are partially compensated by the perspective mapping. These effects result in small defect areas and standard deviations below a potential threshold.

The same applies to the partially missing label and the second misplaced label of set 4. (See Figure 10.) However, in this case the defect areas and the standard deviations are larger than for set 3 because the missing part is more characteristic and the labels are larger. Part of the first misplaced label is not visible, which cannot be compensated by mapping. Here, both features definitely indicate the defect.

Let us now consider a few cases when the proposed method can potentially fail. Figure 11 illustrates the problem faced when the flask is *transparent*. (The images were enhanced by gamma correction for better visibility.) Both good labels are classified as defective because they significantly differ from the reference in the highlighted areas. These differences are caused by labels on the opposite side whose visible parts vary with flask orientation.

reference label good label 1 good label 2

Fig. 11. Reference label and two instances when the method failed.

Figure 12 shows a case when the above mentioned *misplacement compensation* can lead to a failure. After the perspective mapping, the absolute difference image has only minor areas of deviation, and the misplaced label is not detected.

reference label misplaced label absolute difference

Fig. 12. Reference label, undetected misplaced label, and absolute difference. The label images are shown enhanced for better visibility.

In some cases, obtaining *precise mapping* can be a non-trivial problem. Figure 13 provides an example when a large number of the feature matches is not correct because of the fine label texture and the lack of distinct, robust features. The feature points found in the fine texture are neither distinct nor robust, and many of them are mismatched. Thanks to the robustness of the RANSAC-like homography calculation procedure applied by the proposed method, the obtained mapping is nevertheless correct leading the proper recognition of the misplacement defect. Note that such cases are relatively rare as most labels contain a sufficient number of suitable features.

Even when an appropriate mapping of fine-textured reference label is found, *comparing* the reference to the input can be problematic. Figure 14 shows an example of the situation when the thresholded difference image contains a sufficient number of the above-threshold pixels. However, due to the fine texture, the defect is split into a large number of components whose areas are often below the area threshold. This results in the elimination of many small components and the final less distinct defect area.

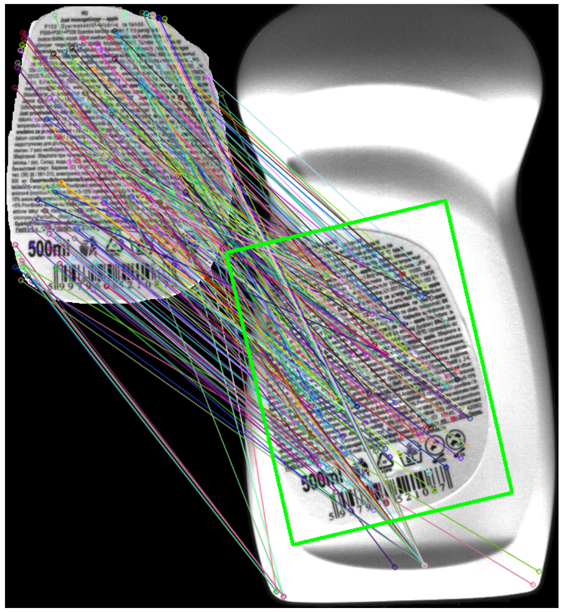
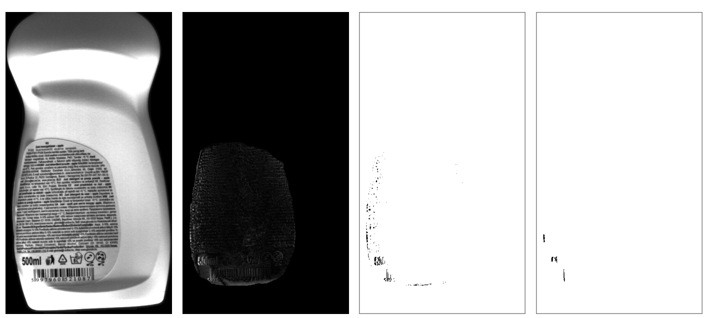


Fig. 13. Example of non-trivial feature matching due to fine texture and lack of distinct features.



input | abs.difference | thresholded | area-filtered

Fig. 14. Example of problematic label comparison due to fine texture.

5 Conclusion and Outlook

In our initial experimental study, we processed images of over 100 flasks of different types and with varying labels. Both good and defective labels were classified. In a great majority of the cases a correct decision was made in acceptable time, that is, the proposed solution is a feasible approach to this industrial application.

However, as discussed in Section 4.1, a few questions are still open. In particular, the following problems need to be addressed in future research and development:

- To indicate the *misplacement compensation* of a rotated or shifted label, one needs to measure the position and orientation of the main axis of the bottle. This can be done by finding the contours of the bottle. For bright flasks, the solution is simple, but the images of dark flasks whose contours are poorly visible may need pre-enhancement.
- Coping with *transparent flasks* also seems to be difficult, and at the moment we do not have clear ideas of how to approach the problem.
- *Wrong mappings* due to featureless labels or labels dominated by fine texture can be indicated by a low number of distinct features whose strength is measured by the extractor. The reliability of the homography estimation provided by the estimator can also facilitate the solution of the problem.
- In the case of labels with very fine texture (tiny letters), selection of an optimal, lower defect area threshold can probably be helpful in *avoiding the removal* of the small parts of a defect region.
- Alternative *measures of discrepancy* between the matched reference and input labels will be considered.
- *Large-scale tests* with much more input data under varying conditions are needed to demonstrate the practical applicability of the proposed method.

Acknowledgments

The study was supported in part by Project no. ED_18-1-2019-0030 (Application domain specific highly reliable IT solutions subprogramme) implemented with the support of the National Research, Development and Innovation Fund of Hungary, financed under the Thematic Excellence Programme funding scheme. The study was also supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00001).

References

1. H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110:346–359, 2008.
2. Kirit Bhatt. Design and implementation of image processing techniques for automatic bottle filling and label checking using raspberry pi: A survey. *International Journal of Advanced Information Science and Technology*, 5:89–92, 2016.
3. J.R. Davies. Method and apparatus for inspecting position of labels on product surfaces, 1975. US Patent 3,891,324.

4. M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981.
5. Enrico Grosso, Andrea Lagorio, and Massimo Tistarelli. Automated quality control of printed flasks and bottles. *Machine Vision and Applications*, 22:269–281, 2011.
6. M.K. Hassan. *A New Approach for Industrial Product Inspection Based on Computer Vision and Image Processing Technique*. PhD thesis, Universiti Teknikal Malaysia Melaka, 2014.
7. Intel Corporation. OpenCV: Open Source Computer Vision Library. `https://sourceforge.net/projects/opencvlibrary/`, 2019.
8. S. Nichani. System or method for detecting defect within a semi-opaque enclosure, 2000. US Patent 6,134,343.
9. D.O. Rehse, A.P. Brooks, and M.J. Bilello. Bottle label detector-label inspecting and sorting apparatus, 1971. US Patent 3,613,885.
10. R.D. Wertz, J.P. Davies, and R.H. Cormack. System for non-contact colored label identification and inspection and method therefor, 1993. US Patent 5,245,399.