Bottle Line Detection using Digital Image Processing with Machine Learning

Anush.Ch¹, K.Yashwanth², S.Shashank³ M.Venkat Reddy⁴ Ashwani Kumar⁵

¹,⁵Department of Computer Science and Engineering, Sreyas Institute of Engineering and Technology, Hyderabad, India.

ashwani.kumarcse@gmail.com

Abstract. Image processing is often used in industrial applications. The automatic visual inspection of a product before it is packaged and dispatched to the client is one of the most common uses for image processing. Using these types of systems in production facilities helps the avoidance of situations where a faulty or sub-standard product is shipped to the customer. The quantity of the fluid in each bottle differs with the accuracy level of the machine and furthermore due to mistakes done by employee during the underlying arrangement. Inappropriate filling during a bottle leads the makers to dispose of whole bunches of an item resulting in huge capital loss. A Economical mechanized review structure for inline packaging which detects oddities has been examined. The system are ending up being to be assign with checking the degree of the fluid filled inside each bottle as the container goes through the assessment framework introduced round the current packaging line. It will help to keep up the product quality and quantity as well and furthermore maintains the accuracy of the filling. The bit of the space between the outside of the fluid inside the bottle and container will be used to observe the differences like underfilling and overfilling. The proposed work is accomplished using image processing techniques in Machine Learning using python platform. Here we use ANN algorithm. Process of detecting the liquid level in bottle as follow as a group of pictures were given, taken under close to consistent lighting conditions in the manufacturing plant. These arrangement of pictures contains both underfilled and overfilled bottles which the packaging organization expects that you should use to identify certain issues as a component of the filling, covering and naming tasks before the final packing stage. The distance for each dataset must be same. We need to provide threshold value to detect if the bottle is overfilled or underfilled.

1. Introduction

Bottling lines are an essential part of food, cosmetics, healthcare and chemical product industries. Real-time monitoring of the bottling line enables the manufacturer to enforce a certain degree of quality control. In this area, accurate filling, inspecting cap closure, sorting recycling plastic bottle, recognition between glass bottle and pet bottles, inspection for overfill or underfill, verification of label quality and detection of defect products are some significant parameters which seem to be necessary to be inspected [1]. Industrial vision systems are not able to handle all tasks in every field of application [2]. Humans as inspectors are slower and cannot keep up with the current speeds of bottling lines [3]. Hence, real time monitoring of bottling plants can be achieved using digital image processing.
processing techniques

Computer vision and machine learning have been utilized in the past to handle a variety of real-world challenges, such as identifying and evaluating things on a screen depending on requirements [4]. As a result, these algorithms may be used for bottle detection as well as object detection [5].

The objective of the research paper is to build a desktop application to find the bottle filling level. In order to detect the bottle level we use ANN algorithm which provide fine accuracy. A combination of segmentation and morphological operations is being used and is explained completely through figures. The results obtained will be uploaded on the cloud for further analysis by the bottling machine manufacturer and bottling plant owners to get further insight into data.

2. Literature Survey

According to the survey authors have presented an Automated Visual Inspection Systems which automatically have a tendency to to make a appropriate decision by checking the quality of the cap closure and detect overfilling/underfilling anomalies [6]. The accuracy of the system is found out to be good but with changes in the bottle shape and size such a system will takes lot of time and computation power in-order to train the neural network again. In the existing system authors have put forth algorithms using which fill level, bottle cap detection, label detection and dirty bottle detection is done, so that a full proof system can be designed for bottle packaging industry [7]. The authors have used computationally light algorithms as compared to the use of neural networks in. The system checks a number of anomalies and offers a complete bottling plant monitoring system but the algorithms are performed on colour images hence more computation time is involved [8]. The algorithms also suggest using static reference lines for feature detection which will again pose a problem in case of change in bottle shape or size.

The authors presented an image processing-based bottle filling and label checking embedded system that can regulate the filling machine until the bottle is perfectly filled. This method necessitates the use of a visual inspection system for each filling nozzle, which will necessitate the integration of the filling gear with the image processing unit [9]. Because the computing time of the algorithm is not given in any of the above-mentioned scenarios, the practicality of such a system in a fast-paced bottling line filling more than 40 bottles per minute cannot be assured [10]. This methods had its own advantages and disadvantages. The main disadvantage being that being an RGB matrix method it could be implemented to detect just three basic colours i.e. red, green, blue. This was a very easy but not a practical solution on basis of industrial application. Our proposed solution differs from the above, as it offers a different algorithm to detect underfill/overfill anomalies within a margin. The system is completely independent of the existing [11-14].

3. Proposed Approach

In the implementation phase all the development activities of the research work is done. The existing system is based on manual processes and is far more difficult to run than the suggested solution. A good implementation is necessary in order to produce a constant system that meets the organization's needs. We are using MLP classifier for suitable classification of predicting the problems where the input are allocated. One or more layers of neurons make up these structures. Data is supplied into the input layer, and predictions are generated on the output layer, sometimes known as the visible layer. There may be one or more hidden layers offering levels of abstraction. In other words, starting with what's needed design takes us towards the way to satisfy the requirements. the planning of a system is probably the foremost critical factor affection is the standard of the software; it's a serious impact on the later phase, particularly testing, maintenance. The output of this phase is that the design document. This document is analogous to a blueprint for the answer and is
employed later during implementation, testing and maintenance. The planning activity is usually divided into two separate phases system design and detailed design.

![Image](image1.png)

**Figure 1.** Basic approach of object detection.

3.1 Object Detection:
For detecting level of liquid in the bottle we use MLP classifier for feedforward artificial neural network that generates a set of outputs from a set of inputs. Object will be detected by using three categories exactly filled, overfilled and underfilled. The image path will be declared. From the path the image will be converted to target size. Then that image is stored in array after dividing the image by 255 we will append the image to data and category to class.

![Image](image2.png)

**Figure 2.** Architecture of ANN algorithm.

3.2 Module description:
This system contains 4 modules.

3.3 Admin
Admin will be able to log in and build the model and detects the image.

3.4 Build model:
Here feature extraction of dataset is completed. We have assigned data to x-train and class to y-train. Now training will be processed by using MLP model. In this x train and y train will be trained.
3.5 Accuracy:
*It* is basically a ratio of accurately anticipated data to total observations, and it represents the quantity of performance.

\[
\text{Accuracy} = \frac{\text{Number of images correctly detected}}{\text{Total no. of images}} \times 100
\]

3.6 Precision:
*It* is defined as the proportion of accurately expected positive observations to the total number of positively expected observations [15].

3.7 Recall:
Recall is defined as the ratio of precisely expected positive observations to all observations in the actual class.

3.8 F1 score:
The F1 Score is calculated using a weighted average of Precision and Recall. So, this score takes into account both false positives and false negatives [16-17].

4. Result Analysis
The requirement specifications from the first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture. The software code to be written in the next stage is created now. In this phase the bottle level will be detected. We are using QPixMap to read file name. The given image will be compared with the trained data image based on the target size. Each input image is compared and output will be generated.

![Activity Diagram of implementation of Bottle level detection.](image-url)
Fig 3 shows the implementation of research paper, which will explain the sequence of actions taken for implementation. Initially admin supposed login and he need to build the model by training the dataset with MLP classifier. Then the bottle inspection is processed with remaining dataset. They are compared with the trained one. The modules we used are:

- Keras is used to maintain aspect ratio of the image. It acts as interface for tensorflow library
- Scikit learn: It includes support vector machines as well as other classification, regression, and clustering methods.
- NumPy provides the support for dealing with the large arrays. This library also provides functions to apply on these arrays.
- SciPy is a Python library which is also open source used for mathematical computations like distance calculation, algebra, and other similar tasks.

![Prediction](image)

**Figure 4.** Line detection using proposed approach.

Figure 4 displays the output after execution. It detects every bottle by comparing with the trained dataset by displaying overfilled, underfilled or exactly filled.

![Algorithm Performance](image)

**Figure 5.** Algorithm Performance.
5. CONCLUSION

In Industry, bottling lines used in the food and medical industries must be continuously monitored in order to create meaningful data. We looked into a general problem with the bottle inspection system, specifically the inspection of the amount of liquid inside the bottle. With the help of a digital image processing algorithm, the proposed system will be able to inspect this parameter. By using such a system, bottling companies will be able to ensure the quality of their services while also being able to remotely monitor their manufacturing lines. Data provided by the system can be scrutinized further to determine equipment health, allowing for prompt repair and a reduction in losses. The system's projected outcomes have been thoroughly discussed.

REFERENCES

[1.] Leila Yazdi, Anton Satria Prabuwono, Ehsan Golkar, “Feature Extraction Algorithm for Fill Level Inspection in Bottling Machine” in International Conference on Pattern Analysis and Intelligent Robotics, 2011, pp.47-52.
[2.] Vinit A. Dave, S. K. Hadia,”Automatic Bottle Filling Inspection System Using Image Processing” Resources, International Journal of Science and Research, vol. 4, issue 4, pp. 1116–1120, April. 2015.
[3.] Varshal Shah, Dr.Kirit Bhatt,”Image Processing Based bottle Filling and Label Checking Using Embedded System”, Resources, International Journal of Innovative Research in Computer and Communication Engineering, vol. 4, issue 5, pp. 9778-9787, May 2016
[4.] Mario Campos, Manuel Ferreira, Teresa Martins, “Inspection of bottle crates in the Beer Industry through Computer Vision”, IEEE International Journal, 2010.
[5.] Domingo Mery and Olaya Medina, “Automated visual inspection of glass bottles using adapted median filtering”, IEEE International Journal, 2010.
[6.] Kumar, A., Design of secure image fusion technique using cloud for privacy-preserving and copyright protection. International Journal of Cloud Applications and Computing (IJCAC), 2019, 9(3): p. 22-36.
[7.] Kumar, A. and S. Srivastava, Object detection system based on convolution neural networks using single shot multi-box detector. Procedia Computer Science, 2020. 171: p. 2610-2617.
[8.] Kumar, A., Z.J. Zhang, and H. Lyu, Object detection in real time based on improved single shot multi-box detector algorithm. EURASIP Journal on Wireless Communications and Networking, 2020. 2020(1): p. 1-18.
[9.] Fatima, S.A., et al. Object recognition and detection in remote sensing images: a comparative study. in 2020 International Conference on Artificial Intelligence and Signal Processing (AISP). 2020. IEEE.
[10.] Kumar, A., S.P. Ghrera, and V. Tyagi, Modified buyer seller watermarking protocol based on discrete wavelet transform and principal component analysis. 2015.
[11.] Y.F. Zhen. Design and Implementation of Vision-Based PET Bottle Packaging Quality Automatic Inspection System. Tsinghua University, 2011.
[12.] B.J. Sun, D.H. Zhou. Fast matching algorithm based on NCC. Transducer and Microsystem Technologies, 2007, 26(9):104-106.
[13.] G.Z. Ma, Z.L. Fang, Z Yao. Performance analysis and comparison of SUSAN edge detector. Modern electronic technology, 2007, (08): 189-191.
[14.] Kumar, A., S.P. Ghrera, and V. Tyagi, An ID-based Secure and Flexible Buyer-seller Watermarking Protocol for Copyright Protection. 2017.
[15.] Kumar, A., S.P. Ghrera, and V. Tyagi, Implementation of wavelet based modified buyer-seller watermarking protocol (BSWP). 2014.