

A Real-time Mobile Notification System for Inventory Stock out Detection using SIFT and RANSAC

<https://doi.org/10.3991/ijim.v14i05.13315>

Yacine Merrad, Mohamed Hadi Habaebi (✉), Md Rafiqul Islam,
Teddy Surya Gunawan
International Islamic University Malaysia (IIUM) Kuala Lumpur, Malaysia
habaebi@iium.edu.my

Abstract—Object detection and tracking is one of the most relevant computer technologies related to computer vision and image processing. It may mean the detection of an object within a frame and classify it (human, animal, vehicle, building, etc) by the use of some algorithms. It may also be the detection of a reference object within different frames (under different angles, different scales, etc.). The applications of the object detection and tracking are numerous; most of them are in the security field. It is also used in our daily life applications, especially in developing and enhancing business management. Inventory or stock management is one of these applications. It is considered to be an important process in warehousing and storage business because it allows for stock in and stock out products control. The stock-out situation, however, is a very serious issue that can be detrimental to the bottom line of any business. It causes an increased risk of lost sales as well as it leads to reduced customer satisfaction and lowered loyalty levels. On this note, a smart solution for stock-out detection in warehouses is proposed in this paper, to automate the process using inventory management software. The proposed method is a machine learning based real-time notification system using the exciting Scale Invariant Feature Transform feature detector (SIFT) and Random Sample Consensus (RANSAC) algorithms. Consequently, the comparative study shows the overall good performance of the system achieving 100% detection accuracy with features' rich model and 90% detection accuracy with features' poor model, indicating the viability of the proposed solution.

Keywords—Computer vision; Inventory management; Object detection and tracking; RANSAC; SIFT.

1 Introduction

Adequate safety stock levels permit business operations to proceed according to their plans. An optimized supply chain will help prevent stockout situations which can always lead to bad decision-making and poor business outcomes [1]. Furthermore, stockout inspection or inventory management is usually done manually by human labor which makes it vulnerable to errors. In order to tackle this issue, many solutions have been provided by business management researchers. While business

management experts have provided human solutions regarding mentoring and analyzing the inventory process [2], engineers also got involved in applying the modern technologies to automate and improve inventory management, using Artificial Intelligent and Internet of Things. Right now, cloud-based inventory systems can track items in real-time. Products usually have either an RFID tag or barcode label, so they can be scanned and identified by the system [3]. Currently, this is how systems can provide visibility into inventory levels, expiration dates, item location, forecast demand, and more [4]. Moreover, Convolutional Neural Network (CNN) was also used in real time mentoring for inventory management [5]. This aims to have an efficient method to count and localize the objects in inventory by utilizing computer vision technology. In the proposed approach, the Connected Component Analysis (CCA) followed by CNN and SoftMax layer is used for object identification and counting for efficient inventory management purposes. Additionally, an information system to detect (thus measure) products in grocery retail sector for example, which are not on the shelf, is also made possible [6]. Accordingly, this has come up with a technical solution for stockout risk too. In their work, authors have proposed a machine learning technique-based approach. This makes use of a rule-based information system developed for a large size retail chain in Greece. The proposed method was compared with the so-called Out Of Stock (OOS) Index approach [6]. The system was able to detect about 27% of the daily occurring OOS cases with accuracy greater than 90%.

In this regard, many works have been achieved on template matching. Chin-Sheng Chen et.al.[7]have analyzed the template matching techniques using statistical models and parametric template for multi-template. This algorithm consists of two phases which are training and matching. In the training phase, the statistical model that is created by Principal Component Analysis method can be used to create a multi-template whereas, in the matching phase, the normalized cross correlation is used to find out the part of an image. In their work, authors have proposed a template matching algorithm based on multi-template using training and matching phases. The Image block and multi-template are built to use the parametric template method. Kavita Ahuja et.al.[8]have analyzed the performance of two template matching algorithms which use the correlation method and phase angle method in order to recognize the object. The goal is to find similar objects when the input is entirely in image form. It is important to notice that the Phase angle method takes very few seconds to recognize the objects in images. Moreover, on rotating the same images, correlation method takes less time for recognizing the same objects. Ankit Kumar et.al.[9]have implemented an approach which focused on the core basics of the template matching applied to remote sensing images using two approaches. They are Sum of Absolute Differences (SAD) and Sum of Squared Differences (SSD). This work finds the so-called Ground Control Point (GCP) and generates the template forms. These templates can run on the image to find the correct location of the temporal image. Duc Thanh Nguyen et.al.[10]have proposed a new template matching method known as Chamfer matching method. It manipulates a Generalized Distance Transform (GDT) and an Orientation Map (OM). The GDT allows weighting the distance transform more on the strong edge points, while OM provides added local

orientation information for matching. Two stages of human detection method such as template matching and Bayesian verification, have been developed. This method can effectively reduce the false positive and false negative detection rate. However, this technique is now old fashioned since it is not scale or angle independent; so, if the same object within the image is scaled up or rotated with a certain angle, the matching will not occur between the template and the object.

This paper proposed a smart solution for out of stock issue in warehouses, by using computer vision technology. It consists of a real-time notification system developed by using machine learning with the exciting (Scale Invariant Feature Transform (SIFT) feature detector [11] and Random Sample Consensus (RANSAC) [12] algorithms. Let's mention that these algorithms are implemented into Matlab environment. Notably, they are able to provide satisfactory performance to tackle the inventory management problem. The structure of the work can be outlined as follows. The first section briefly describes tools and techniques that have served the present study regarding object detection and tracking as a computer technology in computer vision and image processing. The methodological aspects to reach the expected goals, are pointed out in section 2, while the third section spells out in more detail the offered results. Strength and weakness of the method will be discussed too. Finally, some conclusions are provided and some prospects are drawn out.

2 Techniques and Materials

The Scale Invariant Feature Transform algorithm (SIFT) was patented in Canada by the University of British Columbia and published by David Lowe in 1999 [11]. This technique consists of taking the image, then blurring it using a Gaussian filter with increasing values of the standard deviation σ ; the obtained output with the lowest value of σ is then subtracted from its corresponding output with value of σ that is directly above. These operations of subtracting, result in the Laplacian Δ of the Gaussian filtering of the original image with increasing values of σ . This is called the pyramidal Laplacian of Gaussian filtering [13].

$$\frac{\partial G}{\partial \sigma} = \sigma^2 G, \text{Heat Equation} \quad (1)$$

$$\sigma \Delta^2 G = \frac{G(x,y,k\sigma) - G(x,y,\sigma)}{k\sigma - \sigma} \quad (2)$$

$$G(x,y,k\sigma) - G(x,y,\sigma) \approx (k-1)\sigma^2 \Delta^2 G \quad (3)$$

where G is the Gaussian; σ is the standard deviation; Δ is the Laplacian; (x,y) are the pixel coordinates, k is a constant coefficient.

The Laplacian of Gaussian acts as a blob detector. Actually, some features or blobs with different scales can only be detected with a certain value of σ , but may be not with another. This makes the SIFT Features Scale invariant. Notably, Figure 1 describes the main steps of SIFT algorithm. On the same note, Figure 2 illustrates how SIFT technique is able to detect image features.

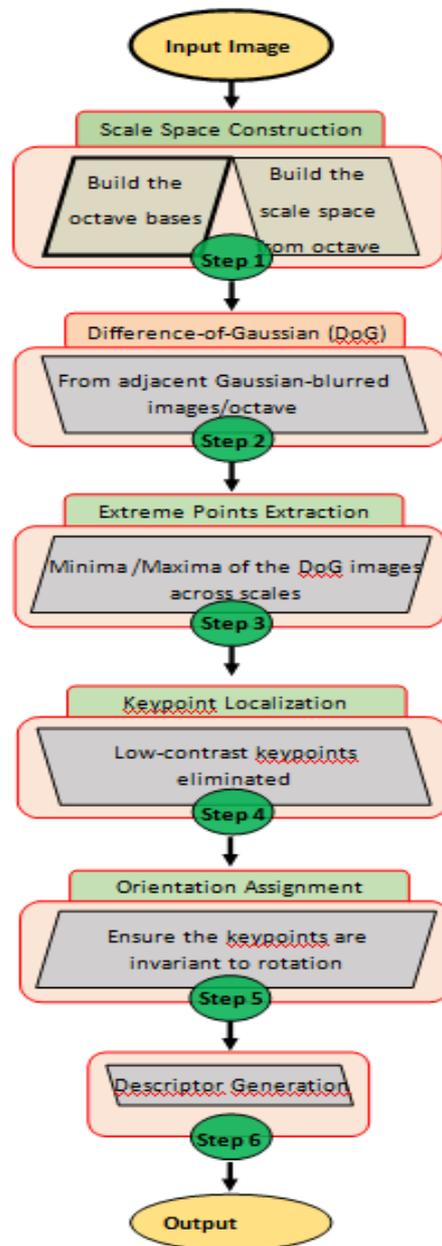


Fig. 1. Processing Pipeline of SIFT

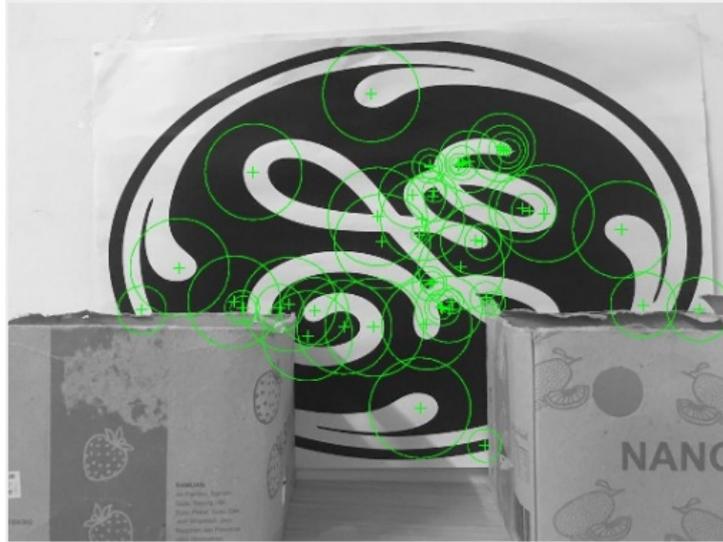


Fig. 2. Feature detected by SIFT technique.

In order to make the feature detector more robust, different scales of the image are used and the pyramidal difference of Gaussians is applied for each scale (octave). These are chosen values of σ which experimentally give the best results [11].

Table 1. Values of σ for different octaves

	Scale →				
Octave	0.707107	1.000000	1.1414214	2.000000	2.828427
	1.1414214	2.000000	2.828427	4.000000	5.656854
	2.828427	4.000000	5.656854	8.000000	11.313708
	5.656854	8.000000	11.313708	16.000000	22.627417

The features that are chosen to be candidates are found using Scale Space Peak detection following the steps below: First, compare a pixel with its 26 neighbors (the 8 direct neighbors + the 18 neighbors from the adjacent scale). Then, select the pixel as a feature if it has the smallest/biggest value compared to its 26 neighbors. Figure 3 serves at illustrating this process.

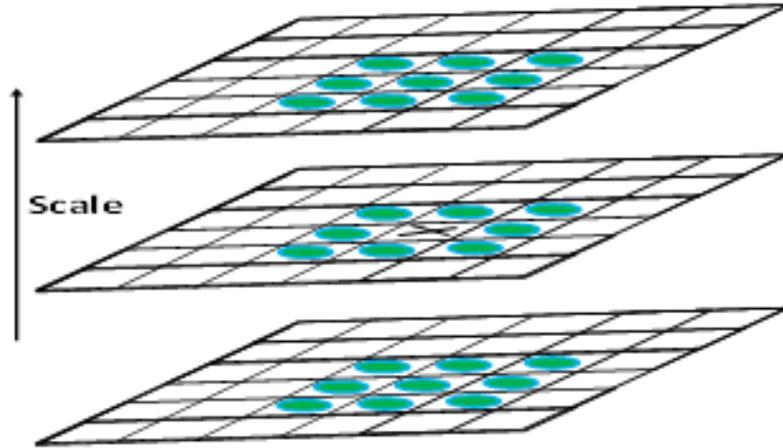


Fig. 3. Comparing the pixel with its 26 neighbors

The responses that represent edges are then eliminated from those candidates using the method for edge detection discussed in the Shi-Tomasi Corner detector [14]. Let us note that this is achieved by computing the eigen values of the Hessian matrix derived from the pixel. Indeed, the pixel does not represent an edge if both the Eigen values of the matrix are bigger than a certain threshold. Once the features are selected, the next step is to assign an orientation to each of them, to achieve rotation invariance. For this purpose, the central derivative, gradient magnitude and angle of the smoothed image are computed as provided in the set of equations below:

$$M(x, y) = (L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2 \quad (4)$$

$$\theta(x, y) = \tan^{-1}((L(x, y + 1) - L(x, y - 1)) / (L(x + 1, y) - L(x - 1, y))) \quad (5)$$

where M is the Gradient magnitude of the smoothed image; L is the Laplacian Gaussian; θ is the angle of the smoothed image.

A weighted direction histogram (as in Fig4) is then created in a neighborhood of a key point using typically 36 bins (36 levels of quantization of the angle); the weight is the magnitude of the gradient. So, the angle with the highest weight will represent the direction of the feature. After the key points are selected, each one is then assigned an orientation. The next step is to assign a descriptor to each key point. The descriptor provides information about the key point neighbors; it represents the histogram of the gradient orientation around the key point within a certain window.

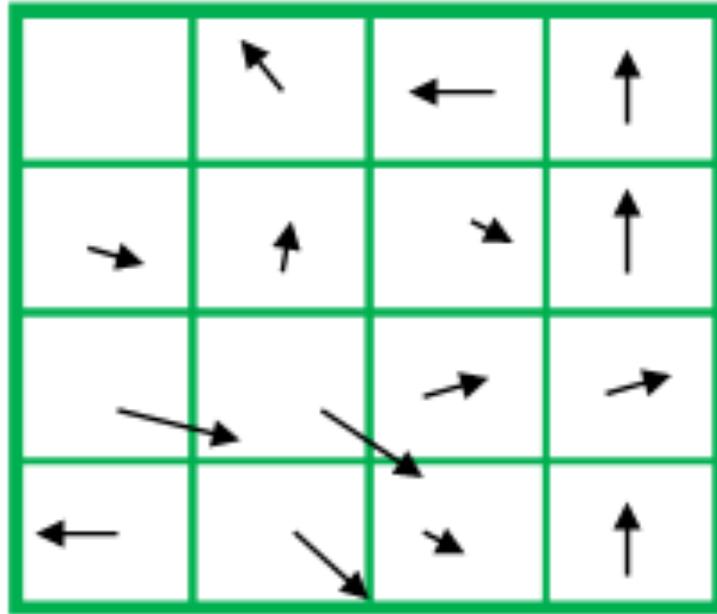


Fig. 4. Weighted direction histogram

At this stage, we can match each key point from the image with its most similar one from another image. Sometimes the algorithm results with some incorrect matches where the two key points are similar but does not belong to the same object; this kind of key points are called outliers. In order to eliminate the outliers to keep the inliers only (ideally), the so called RANSAC algorithm (Random Sample Consensus) is used [12]. The key idea is to find the best partition of points in inlier and outlier sets, then estimate the model from the inlier set. The algorithm consists of the following steps:

Algorithm: Processing RANSAC
Step1: Sample the number of data points required to fit the model.
Step2: Compute model parameters using the sampled data points.
Step3: Score the fraction of inliers over outliers within the present model.
Step4: Repeat steps 1 to 3 until find the best model.
<i>End.</i>

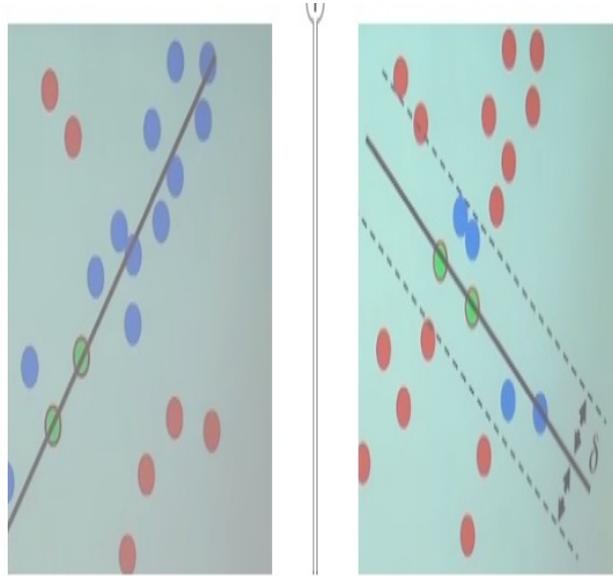


Fig. 5. Defining inliers and outliers with respect to two different lines

We can clearly see that the model in the left side of Figure5 counts a larger number score of inliers within the chosen range, than the model that is given in the right side of this same Figure.

3 Methodology

The present work is basically devoted to achieving a real time out of stock detection in inventories, using the previously discussed object detection techniques. This is achieved in such an accessible manner that improves the understanding process, and pointing out the strengths of the provided solution. The products should be stocked according to their contents in the shelves' rows. Each row should be containing boxes of same products. The idea is to tag in the back of each row, the Logo of the particular product it is stocked in. The Tag should spam the whole area of the row. Initially, the rows are full and the Logo is hidden by the cardboard boxes; when the boxes are taken out of the row for delivery, the logo will be gradually visible, until it can be detected by the inventory cameras. Smart cameras are required. They must be able to detect the logos of the products stored in the inventory by implementing the Object detection algorithm using the SIFT (Scale Invariant Feature Transform) feature detector and RANSAC. Figure 6 is provided hereafter for an easy follow-up of the method.

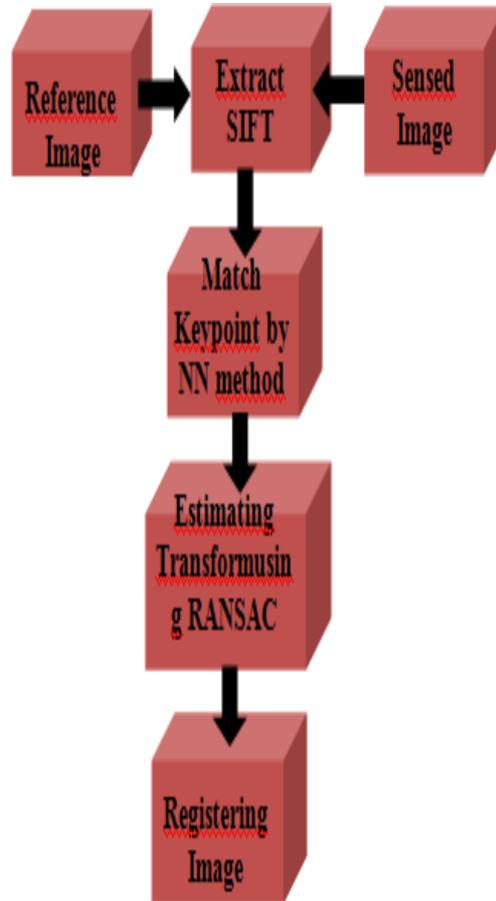


Fig. 6. SIFT and RANSAC process

Once a logo has been detected by one of the cameras, which is connected to a microcontroller, it must be then identified. Accordingly, the microcontroller should send an out of stock query to the central computer. Each Logo should be linked to a code which is in turn linked to a product type and brand. The query should notify to the central computer, the code of the logo that has been detected, so that the product which is out of stock is made identified.

The most appealing features in this technique are that it is a low cost, low tech. As a matter of fact, it makes use of a relatively simple object detection algorithm which does not require high processing and storage power. Moreover, it requires Closed-circuit television (CCTV) cameras that are necessarily deployed in every inventory. Hence, the accuracy of the technique that is presented in this work, will be assessed according to three different parameters. The first parameter would be the type of logo and its choice. The second parameter will be the angle in which the camera is placed according to the row. This should enlighten on the number and position of cameras to

be added for an optimum Prize/Accuracy ratio. The aim is to deploy the smaller number of cameras while keeping a high accuracy rate.

To sum up, the process of Inventory stock-out detection of the present work, using SIFT and Ransac object detection is depicted in Figure 7.

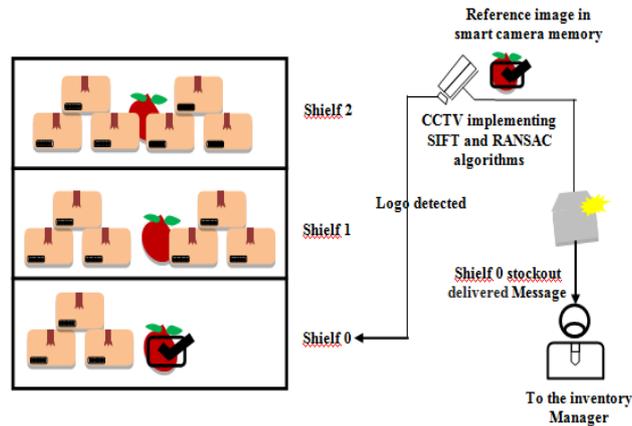


Fig. 7. Inventory stock-out detection using Sift and Ransac object detection

4 Experiments and Results

In this section, the obtained results will be discussed from conducting our own real time experiments. Two different logos have served this study. The first one was the Apple brand logo and the second, was the General Electric brand logo. Then, we categorized the first logo in what we named a features' poor logo category, where the second was categorized as a features' rich logo. The characterization was based on the number of SIFT features detected in each logo. Twenty (20) features have been set as being the threshold number of features separating the two categories.

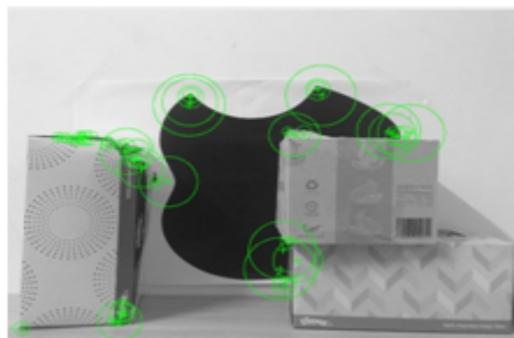


Fig. 8. Features detected in the first Logo

The first experiment was performed on each logo separately. It aims at assessing the accuracy of the detection based on the Logo choice. The logos were printed in an A3 paper format, spamming all the paper. Each logo was then glued on a shelf row. Then, each of them was hidden, using cardboard of the same sizes. Each card board covers 1/9 of the logo. Initially nine boxes were deployed to cover the entire logo. Then the algorithm using Matlab is run, in order to detect the logo. For this purpose, we first use a picture of the logo, naked as a reference and then, we try to detect the logo on images which are taken after removing one of the boxes, each time. It is assumed that the algorithm works with a no fail accuracy if the logo is detected only when more than 7/9 of it, is visible.



Fig. 9. Matching separated Logos with reference images.



Fig. 10. Logos detected perfectly in the first experiment

The obtained results are provided in Figures8-10. The second experiment was also conducted on each logo separately using the same parameters as those used in the previous case, except that another angle is used for capturing the images. The objective is to point out the impact of the angle of the camera with respect to the shelf and rows.

The Last experiment will be conducted on both logos simultaneously. They would have been deployed one next to the other in a row, and each of them would be hidden

by nine boxes. The algorithm will try to detect the logo that would be visible. If both are, the algorithm should be able to detect the two of them. We keep the same reliability criterion as in the previous experiments.



Fig. 11. Matching the first Logo from both Logos simultaneously



Fig. 12. Detecting the first Logo from both Logos simultaneously

The flowchart of Figure 13 illustrates the obtained results for each simulated scenario. Accordingly, the logo choice has an impact on the stock out detection reliability, since the features' rich logo showed a 100% accuracy, whereas the features' poor logo showed a slight less performance with an accuracy of 90%. Both used angles showed similar results. The last experiment exhibits a lack of robustness in the algorithm. It showed an accuracy of 80% on the features' poor logo and was completely defective in detecting the features' rich logo. This is due to a default in the defining of the outliers from the inliers policy, used by RANSAC algorithm and can be overcome by an appropriate setting of the parameters.

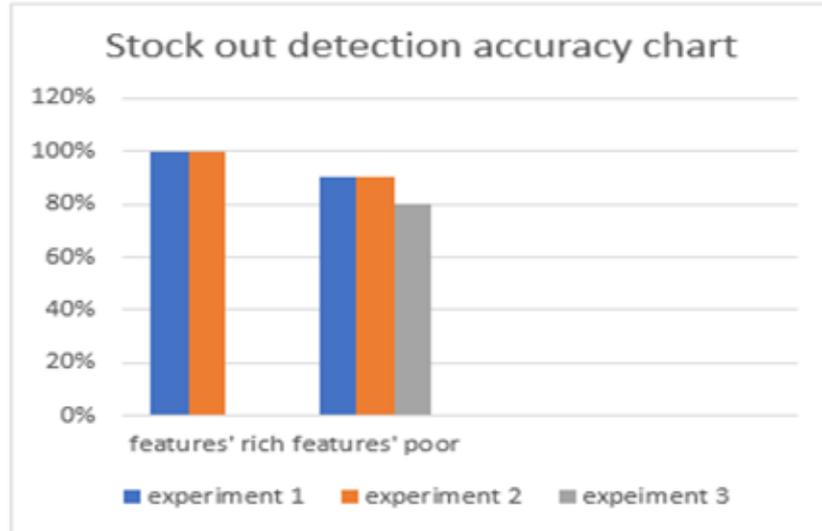


Fig. 13. Stockout detection accuracy

5 Conclusion

This work has demonstrated that the combination of SIFT and RANSAC algorithms is a reliable and efficient object detection solution for inventory stock out process. Consequently, this leads to increased business outcomes and good decision making, depending on efficient products ordering and accurate data accumulated by the proposed system. Although, these algorithms have been developed long time ago, but they still prove to be effective when used in an effective manner. Furthermore, they are economically justifiable and do not consume system resources and thus, they are considered to be easily implemented. Finally, some remarks could be drawn out for future work and prospects. Accordingly, RANSAC parameters should be adapted in such a way any feature outside the logo borders is considered to be an outlier.

6 Acknowledgement

This work was conducted in IoT and Wireless Communication Protocols Laboratory and is partially funded by the International Islamic University Malaysia (IIUM) Publication-Research Initiative Grant Scheme No. P-RIGS18-003-0003.

7 References

- [1] H. L. Lee and C. J. S. m. r. Billington, "Managing supply chain inventory: pitfalls and opportunities," vol. 33, no. 3, pp. 65-73, 1992.

- [2] D. G. Bauer, R. J. Campero, P. B. Rasband, and M. D. Weel, "Inventory management system," ed: Google Patents, 2012.
- [3] J. F. Kimball and S. B. Leonard, "Inventory management system using RFID," ed: Google Patents, 2010.
- [4] B. S. S. Tejesh and S. J. A. E. J. Neeraja, "Warehouse inventory management system using IoT and open source framework," 2018. <https://doi.org/10.1016/j.aej.2018.02.003>
- [5] N. K. Verma, T. Sharma, S. D. Rajurkar, and A. Salour, "Object identification for inventory management using convolutional neural network," in Applied Imagery Pattern Recognition Workshop (AIPR), 2016 IEEE, 2016, pp. 1-6: IEEE. <https://doi.org/10.1109/AIPR.2016.8010578>
- [6] D. Papa Kiriakopoulos, K. Premature, and G. J. D. S. S. Doukidis, "A decision support system for detecting products missing from the shelf based on heuristic rules," vol. 46, no. 3, pp. 685-694, 2009. <https://doi.org/10.1016/j.dss.2008.11.004>
- [7] J.-J. H. Chin-Sheng Chen, Chien-Liang (August 2013). Template Matching using Statistical Model and Parametric Template for Multi-Template. Available: <http://www.scirp.org/journal/jsip>
- [8] K. A. e. al. (March 2013). Object Recognition by Template Matching Using Correlations and Phase Angle Method. Available: www.ijarce.com
- [9] A. Kumar, A. Joshi, A. Kumar, A. Mittal, D. J. I. J. o. S. P. Gangodkar, Image Processing, and P. Recognition, "Template matching application in geo-referencing of remote sensing temporal image," vol. 7, no. 2, pp. 201-210, 2014. <https://doi.org/10.14257/ijsp.2014.7.2.19>
- [10] D. T. Nguyen, W. Li, and P. O. J. P. R. Ogunbona, "Human detection from images and videos: A survey," vol. 51, pp. 148-175, 2016. <https://doi.org/10.1016/j.patcog.2015.08.027>
- [11] D. G. Lowe, "Object recognition from local scale-invariant features," in Computer vision, 1999. The proceedings of the seventh IEEE international conference on, 1999, vol. 2, pp. 1150-1157: Ieee. <https://doi.org/10.1109/ICCV.1999.790410>
- [12] M. A. Fischler and R. C. J. C. o. t. A. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," vol. 24, no. 6, pp. 381-395, 1981. <https://doi.org/10.1145/358669.358692>
- [13] H. Kong, S. E. Sarma, and F. J. I. T. o. I. T. S. Tang, "Generalizing Laplacian of Gaussian filters for vanishing-point detection," vol. 14, no. 1, pp. 408-418, 2013. <https://doi.org/10.1109/TITS.2012.2216878>
- [14] C. S. Kenney, M. Zuliani, and B. Manjunath, "An axiomatic approach to corner detection," in null, 2005, pp. 191-197: IEEE.
- [15] E. Mirsadeghi, K. Faez, M. Fotouhi, and S. Kasaei, "Feature-based image stitching with repetitive patterns using local phase quantization," in Electrical Engineering (ICEE), 2014 22nd Iranian Conference on, 2014, pp. 1147-1151: IEEE. <https://doi.org/10.1109/IranianCEE.2014.6999708>

8 Authors

Yacine Merrad is pursuing his Master of Science in Computer and Information Engineering at the International Islamic University Malaysia. His research interests are towards the fields of Telecommunications, Internet of Things, Sensor Networks, Data Analytics and Electronics. Email: yacinemerrad10@gmail.com

Mohamed Hadi Habaebi is a professor with the department of Electrical and Computer Engineering, International Islamic University Malaysia. His research interests are in IoT, wireless communications, small antenna design, wireless channel propagation, FSO propagation and Networking. Email: habaebi@iium.edu.my

Md Rafiqul Islam is a professor with the department of Electrical and Computer Engineering, International Islamic University Malaysia. His research interests are in wireless channel modeling, radio link design, RF propagation measurement and modeling in tropical and desert, RF design, smart antennas and array antennas design, FSO propagation and modeling etc. Email: rafiq@iium.edu.my

Teddy Surya Gunawan is a professor with the department of Electrical and Computer Engineering, International Islamic University Malaysia. His research interests are in embedded system design, speech and image processing, AI and security. Email: tsgunawan@iium.edu.my

Article submitted 2020-01-16. Resubmitted 2020-02-25. Final acceptance 2020-02-27. Final version published as submitted by the authors.