HMEDASYS: HOME MEDICATION INTAKE ALERT SYSTEM

Jardson Rodrigues[1], Christophe Soares[1,2], Pedro Sobral[1,2], Rui S. Moreira[1,2] and José M. Torres[1,2]
[1]*ISUS Unit, Faculdade de Ciência e Tecnologia, Universidade Fernando Pessoa, Porto, Portugal,*
[2]*Artificial Intelligence and Computer Science Laboratory, LIACC, University of Porto, Porto, Portugal*

ABSTRACT

The use of medication as treatment is often crucial to improve a patient's health. This process involves several factors, and often patients forgot to take their medicine. Thus, mechanisms that can contribute to therapy adherence are fundamental. The goal of this work has been the development of a real-time and automatic video object recognition system (HMEDASYS) to be used in the context of adherence and medication taking in-home care. The methodology followed has been to train a classifier model for medication pack recognition using Deep Learning. Combining and organizing data from previously annotated images, we retrained a neural network using the YoloV4-tiny Framework. In the experiments developed, two different classes of medicines were used for training and the system achieved a mAP of 98.33% on the test set. During the operation of HMEDASYS, the medication packs are real-time monitored using a video camera coupled with the developed system. In further 8 evaluation scenarios of the operation, only two errors were verified in a total of 16 medication packs to be identified.

KEYWORDS

Drug Recognition, Medicine Recognition, Deep-Learning, Computer Vision, YOLO

1. INTRODUCTION

Adherence to drug treatment is a very important factor for therapy to achieve the expected effect and result. This success includes not only taking medications at the prescribed times and days but also changing some habits (Lam & Fresco, 2015; Osterberg & Blaschke, 2005). Poor adherence to treatment can cause the progression of disease due to incorrect use of therapy. The more adherent the patient is, the better for his health (McDonnell & Jacobs, 2002; Mrosek et al., 2015). With the continuous evolution of technology, the use of Artificial Intelligence (AI) is becoming fundamental in daily tasks. Specifically, in the context of medication recognition, there is still a lack of low-cost, open source, solutions that can alert a patient with chronic or non-chronic diseases. This may assist the patient who takes multiple medications with remainders at a certain time based on the medication's schedule. Other solutions have been reported, but they require a large apparatus of equipment or are not fully autonomous.

This paper proposes a solution that consists of a system based on computer vision, using neural network architectures, with a low computational cost. We developed a software solution capable of assisting people to take their medication. The application works by monitoring the medication in real-time using a simple webcam. The problem addressed in this article consists of managing the schedules and taking medication. The most common problems identified are the discontinuation of treatment and the lack of professional monitoring, or the intrinsic complexity whenever the patients have to intake several medications many times a day. The use of cell phone reminders can help, but they cannot identify which medication has been taken. Therefore, a mechanism that incorporates the vision and identification of the medications on the table, time management, and identification of which medication needs to be taken, can considerably reduce the problems of low adherence. Our major contribution consisted of the creation of a prototype that allows the management of medication taking through computer vision using a low-cost system. The rest of the paper is organized as follows. Section 2 presents state-of-the-art as well as a collection of meaningful related works. The HMEDASYS system developed is presented in section 3 and describes the tools and techniques applied in this project. Section 4 presents an additional evaluation framework adopted and the results obtained. Finally, section 5 concludes this paper, also pointing to some directions in a near future.

2. RELATED WORK

In the first part of this section, some relevant medication take systems and solutions are mentioned and analyzed. The remainder of the section is reserved for the presentation of relevant computer vision, machine learning, and object recognition techniques and software libraries.

2.1 Medication Take Related Systems

One way to decrease medication-taking errors, especially for users without professional supervision, is the use of a medication-taking mechanism. This type of device can help with tasks that are prone to error. Several features are typical in these smart medication adherence products (MAP) as mentioned in (Aldeer et al., 2018; Faisal et al., 2021).

The PillStation system (Bear & Jain, 2011)provides a visualization of the pills that the patient must take over a certain period. The device uses the output of a camera, CCD image sensor, or similar sensor and image capture device focused on the interior space of a storage compartment of the device. An image capture device is positioned to capture an image of the interior space of each compartment, and a communication module transmits the captured image to a central monitoring station.

In (Wehba et al., 2014), the author designed a management system that includes multiple computers, including a medication management unit (MMU) the device is associated with a physician to place a medication order. The medication-by-management system is flexible and integrated to provide increased patient safety and caregiver productivity with a variety of hospital information (Wehba et al., 2014).

The eMar *(Electronic Medication Administration Record)* barcode technology incorporates various technologies into the nursing staff workflow to ensure that the medication is administered at the correct time and the correct dose and to the correct patient. In observations, in contexts without using eMar technology, there were 776 medication administration errors, a rate of 11.5% while in medical units using the technology 495 medication administration errors were observed, an error rate of 6.8%, which represents a 41.4% reduction. Medication administration errors out of schedule dropped from 3.1% to 1.6%, representing a 50.8 percent reduction (Poon et al., 2010).

A medication management system was proposed with a customized box for pills, a pill dispensing unit, software for automatic medication dispensing, and a recognition model for pills. A camera has been integrated into the system to acquire images of pills, which are transferred to the application for recognition of the *CNN Convolutional neural network*. Application software was developed for use with a smartphone with basic functions such as timing and counters. The dispensing of pills requires only the setting of their mixture during the initial parameter setting. The system can inform the taker when to take the medication, and according to the set parameters, the pills are dispensed separately. The need to compare with database images when using pill recognition resulted in a longer processing time (Tsai et al., 2020).

2.2 Object Recognition and Computer Vision

Computer vision has a wide variety of applications such as industrial, self-driving vehicles, surveillance, and others. It deals with varied areas such as computer interaction, image retrieval in digital libraries, medical image analysis, and rendering of synthetic scenes in computer graphics (Forsyth & Ponce, 2003). One of the most important areas is object detection in images or videos (Pop et al., 2017; Szeliski, 2022; Zou et al., 2019).

2.2.1 OpenCV Software Library

The *Open Source Computer Vision Library* (OpenCV) is a cross-platform, computer vision library. The library is written originally in *C* and *C++* and runs on multiple platforms, such as Windows, Linux, Android, macOS, FreeBSD, and OpenBSD, including embedded devices with CPUs based on ARM architecture. OpenCV is designed for computational efficiency and with a strong focus on real-time applications. One of the goals of OpenCV is to provide a simple computer vision infrastructure that helps people create very sophisticated vision applications. In some cases, high-level features in the library will be sufficient to solve complex computer vision problems (Bradski & Kaehler, 2008; Kaehler & Bradski, 2016).

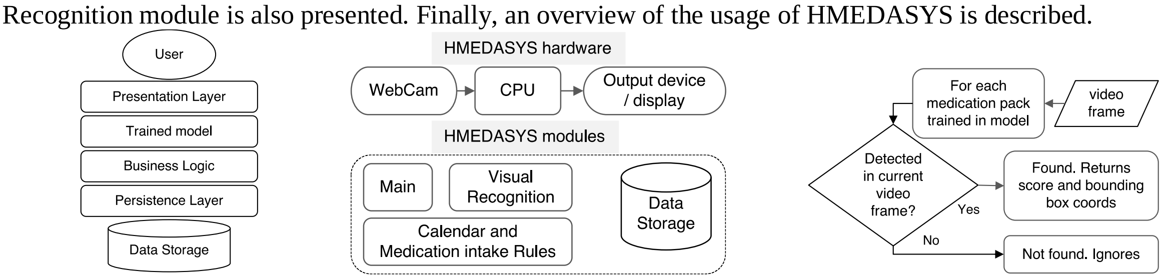
2.2.2 YOLO and Darknet Software Framework

Darknet is a high-performance open-source framework for implementing neural networks and is used, for instance, to implement the object detector YOLO. Darknet is written in C and CUDA and can be integrated easily into CPUs and GPUs processing (Redmon, 2013). Advanced implementations of deep neural networks can be done using Darknet. These implementations include You Only Look Once (YOLO) for real-time visual object detection, ImageNet classification, and recurrent neural networks (RNN). Yolo or *You Only Look Once*, is an algorithm that detects and recognizes multiple objects in an image in real-time. Object detection in YOLOv4 is done as a regression problem and provides the probabilities of categories of the detected images. The architecture of YOLOv4 is composed of CSPDarknet53 as a backbone (Wang et al., 2020), an additional spatial pyramid pooling module, and a new backbone that can enhance the learning capability of CNN. The spatial pyramid pooling block is added to the CSPDarknet53 to increase the receptive field and separate the most significant context resources. Instead of pyramid networks of features for object detection used in YOLOv3, PANet is used as the method for aggregation and is mainly incorporated into the model to enhance the parameter segmentation process for different detector levels. Single-stage models (Bochkovskiy et al., 2020) allow multiple predictions on the same object in a single image, ambiguities are removed from these predictions and subsequently by a process called non-maximal suppression *NMS*, which only leaves the bounding box with higher probability and label for the object (Dazlee et al., 2022). The YOLOv4-tiny detector is proposed based on YOLOv4 to simplify the network structure and reduce parameters, which makes it suitable for development on mobile and embedded devices and improves real-time object detection (Jiang et al., 2020).

3. THE HMEDASYS SYSTEM

The HMEDASYS prototype described intends to contribute to the home healthcare area, providing the capability to detect and recognize medications and trigger alerts to users for medication intake. This system was trained to recognize a predefined number of different medications. In the experiments described in this document, the number of different medication packs trained to be recognized has been, NUM_MEDICATIONS = 2. One important requirement in development of the HMEDASYS was the possibility to use it on low-cost computer platforms, such as Raspberry Pi (Raspberry Pi Foundation, 2022).

The rest of this section presents an architectural overview of the HMEDASYS, the main elements that make up the system architecture for software implementation. The process of training the core model of the Visual Recognition module is also presented. Finally, an overview of the usage of HMEDASYS is described.



(a) HMEDASYS Layered Architecture (b) HMEDASYS Architecture: hardware and software (c) Visual Recognition Module of HMEDASYS

Figure 1. HMEDASYS Architecture

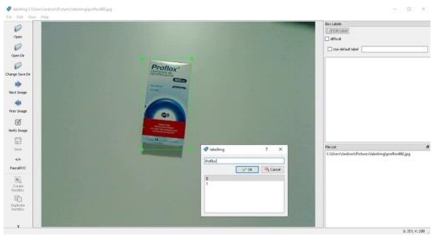
3.1 HMEDASYS Architecture

The HMEDASYS architecture can be organized through a set of software layers that aim to cover the major technical aspects relevant to its operation, as can be seen in Figure 1(a). The presentation layer, at the top, corresponds to the display of an interactive screen with the medicine intake and the alert messages. It is also possible to visualize in this layer the medicines that are being monitored. This presentation layer is detailed in section 3.3. In the Trained Model layer, it is done the recognition of the trained visual object types in the image frames that are acquired through the Webcam, corresponds, in a certain sense, to the Visual recognition Module. The description of the dataset and the process of model training are done in section 3.2. In the Business layer, calendar and medication intake rules are applied that consider three aspects: time information; the existence of the medications on the table or webcam visible surface; and the medical prescription for taking the medications. These rules are used to control the HMEDASYS alerts and outputs. In the Persistence layer, the storage of information on the recognized medications is saved in the Database along with medical prescription information. The implemented prototype offers a solution for monitoring and controlling the medication intake at home using a simple hardware setting (top of Figure 1 (b)). The central idea is to detect and recognize the medicine and to alert its taking to user, based on information stored in the local database. The HMEDASYS software modules are depicted in Figure 1 (b) bottom. The database (Data Storage) is used to store information about medication dose and hours of taking. During operation, the Visual Recognition module of HMEDASYS is constantly searching, in each video frame from the webcam, the occurrence of the trained medication packs in the model so is doing inference. If a visual object is detected with a confidence score greater than the predefined threshold (a default threshold of 0.25 has been used in the tests) then its bound box, label ID, and score are returned to the main module, see Figure 1 (c). One function of the Main module is to store, in a log file, the time intervals when each of the trained medications has been recognized in the viewpoint of the webcam, using the output of the Visual Recognition Module. This log file can be used by the Rule module to produce reports by the HMEDASYS system.

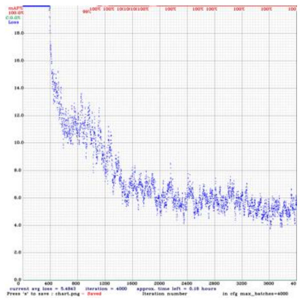
3.2 Dataset Creation and Model Training

For the training of the model, we created a dataset with labeled images of two sizes, 150×220 and 400×500, to improve the recognition and its invariance to image size. In the training stage, a total of 293 images were used: 205 images for training, 59 for validation, and 29 for testing. The relevant visual objects, the medication packs, were annotated as belonging to one of two possible classes (*NUM_MEDICATIONS = 2* as stated before). In Figure 2(a) it is possible to see a sample composed of 6 images of the used dataset, three of each class trained for the experiments. Each selected class represents a specific medication pack to be recognized by the model. In Figure 2(b) one relevant visual object is annotated using the LabelImg software (Tzutalin, 2015).

(a) Dataset sample of 6 images of the two trained classes: Class 1 (top row); class 2 (bottom row)



(b) Dataset annotation of ground truth using LabelImg



(c) Accuracy during training

Figure 2. Dataset creation and training

Using transfer learning, instead of training the model from scratch, allowed to use of the pre-trained YOLOv4-Tiny weights up to the first 29 convolutional layers. After a total of 4000 iterations, the model was trained for the detection and identification of the medicine. The accuracy graph can be seen in Figure 2(c). The mean Average Precision (mAP) obtained in the training was 98.33%.

3.3 Usage of HMEDASYS

3.3.1 User Profile

The typical user of HMEDASYS has diseases and takes several medications or goes through treatment so it is crucial to take the correct medication. This patient does not benefit from the presence of a healthcare professional and remains alone most of the day or night. In his house, he keeps all the medicines he needs on a table or a visible place. A Webcam is positioned over the medications. An alert is issued as a reminder to take the medication. When, for a short interval, the user removes the medication from the table, at the time of taking the medication, a dosing movement is counted, and the alert reminder until the next dose is deactivated.

3.3.2 Design and Usability

The HMEDASYS Medication Alert System graphical user interface (GUI) for the home user is composed of the main window divided into two panels. On the left panel, it is possible to visualize the medications recognized by the system and the webcam image. On the right panel, in the message and textual information area, it is possible to check the status information of the system, time of the day, and alerts triggered (Figure 3(a)). Two major behavior states were considered for HMEDASYS: i) behavior between medication intakes; ii) behavior at medication intake times.

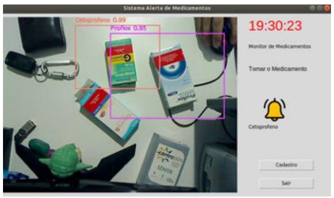
HMEDASYS behavior between medication intakes

Considering all the medication packs previously trained, the system is constantly identifying and drawing a rectangle on the detected visual objects and comparing those with the database of prescribed medicines. On the right panel displays an Ok status message, the green color icon, in case everything all the prescribed medicines are visible, as can be seen in Figure 3(a). The system issues an alert if the prescribed medicine is not positioned adequately, informing the patient that a certain medicine is missing from the table. The system informs which medicine is missing (Figure 3(b)).

(a) Main Window of the HMEDASYS

(b) One Medication not identified

(c) All Medications Missing in scene



(d) Alert for the Medication Intake

(e) One Medication missing

Figure 3. One Medication not identified in the scene

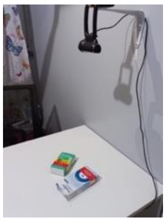
In the cases where the system does not find any of the trained medicines in the scene being monitored, an alert is triggered along with the names of the medicines missing. Until the adjustment is made, the system will continue with the alert, as can be seen in Figure 3(c).

HMEDASYS behavior at medication intake times

The system always monitors the recognized medication packs, the current time, and the stored time of taking each of the prescribed medications. When it is time to take a particular medication, an alert is triggered with an image of a bell informing which medication needs to be taken, as shown in Figure 3(d). The medication intake alert remains on for a certain, configurable, period. During this period, the alert continues, with the bell being displayed until the moment the medication pack is removed from the scene. After the removal of the pack, the system changes the status to Medicine Missing, and informs the name of the medicine, suggesting that after taking the medicine it should be placed back on the table again to proceed with monitoring until the next medication intake time scheduled. After the removal of the medication, the system will identify that there is a certain missing medication that needs to be monitored, this is necessary for the monitoring to continue for the next scheduled time of medication taking for that patient (Figure 3(e)).

4. EVALUATION AND RESULTS

The experiments and evaluations performed are intended to verify the functionality and performance of the proposed system, as well as the representation of object recognition using machine learning. The prototype evaluated was developed using the Python language. The architecture selected for the recognition was Yolov4-Tiny, due to its lightness and ability to be deployed to other platforms, namely, low-cost single-board computers, such as a Raspberry Pi 4 (Raspberry Pi Foundation, 2022). Yet, in the experiments, the components selected for the tests of the Monitoring Module were the following: i) Image acquisition: USB 2.0 Webcam, Logitech Quick Cam Pro 9000 with autofocus; ii) Computer: Intel I5 4590, 4th gen, 8GB RAM, 120GB SSD, OS Ubuntu 18.04.5 LTS and Nvidia Video Card with 2 GB RAM; iii) Display: 18.5” LCD monitor for interaction with the display and information system.



(i) Webcam Setup 1 (optical axis perpendicular to table plane, distance = 40cm)

(a) Scenario 1: Two known Medication packs over the table (one of each trained class)

(b) Scenario 2: Two known and three unknown (untrained) medication packs over the table

(c) Scenario 3: Two known medications packs and low light conditions

(d) Scenario 4: Two known and three unknown (untrained) medication packs over the table and low light conditions

Figure 4. Evaluations Scenarios 1-4

4.1 Evaluation Setup

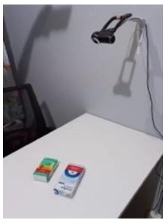
The system has been deployed only, with few resources and easy installation. The experiment consisted in choosing a suitable place for the installation and observing the height at which the camera needed to be fixed, to obtain a better field of view and with little distance from the objects of interest. A Webcam on the table, where the previously trained medication packs are, with a height of up to 40 centimeters, which is adequate for the system to work. This height value used during the experiments is like the one used during the dataset creation for the training stage. This setting can be observed in Figure 4(i) and Figure 5(ii) for the two camera orientations tested. The experiments described in this section were conducted indoors with the presence of natural and artificial light. The objective has been to test the recognition of the two trained visual objects among the objects left randomly over the table. One clear limitation of the approach tested in the experiments is its inability to recognize visual objects occluded or stacked, or only partially visible to the viewpoint of the camera. For that reason, we considered that all the medications are scattered randomly, but visible to the webcam.

4.2 Evaluations: Camera Optical Axis Perpendicular or 90º (Scenarios 1,2,3,4)

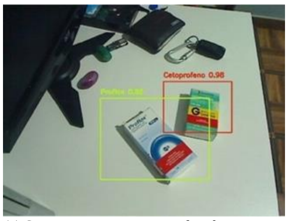
In the environment for this evaluation round, the webcam was positioned at a height of 40 centimeters above the medicines, as can be seen in Figure 4(a-d). The medicines are in the center of the image acquired by the webcam. Performed indoors, with artificial lighting, with a 9 W LED bulb, presenting only the trained medicines on the table, and in a second moment, other medicines were added.

4.3 Evaluation: Camera Optical Axis At 45º (Scenarios 5,6,7,8)

In this evaluation, the Webcam was positioned at a height of 40 centimeters above the medications, but, this time the Webcam was repositioned at a 45-degree angle. The medicines remain in the center of the image, but with a slight inclination. Performed indoors, with the same conditions as in the evaluation from section 4.2 as seen in Figure 5(ii). The trained medicines were positioned at first with few objects, then new medicines were added to simulate an environment with an increased level of difficulty in the recognition, see Figure 5(a-d).



(ii) Webcam Setup 2 (45º angle between the optical axis and table plane, distance = 40cm)



(a) Scenario 5: same as 1 and webcam at 45- degree angle

(b) Scenario 6: same as 2 and webcam at 45- degree angle

(c) Scenario 7: same as 3 and webcam at 45- degree angle

(d) Scenario 8: same as 4 and webcam at 45- degree angle

Figure 5. Evaluations Scenarios 5-8

4.4 Results

The visual recognition module of HMEDASYS was submitted to eight tests for recognizing medicines. Results can be seen in Table 1. Even with adverse situations, in the evaluation set performed in section 4.2, the system was able to recognize 100% of the medication packs trained. In the evaluation set from section 4.3, two failures were observed, one pack erroneously identified (scenario 6) and one pack that was not identified (scenario 8). In a certain sense the HMEDASYS system has a robust functionality that, in case the medication pack cannot be recognized, even if it is on the table, a *Missing Medication* warning is issued, along with the name of the medication as seen in Figure 3(b). In all cases, the alarm, according to the calendar, is triggered.

Table 1. Results of visual identification tests of the two trained Medication Packs

Scenario	Cam axis	Light	Packs over the table	Packs correctly identified	Packs erroneously identified	Error
1	90	normal	2	2	0	0
2	90	normal	5	2	0	0
3	90	low	2	2	0	0
4	90	low	5	2	0	0
5	45	normal	2	2	0	0
6	45	normal	5	2	1	1
7	45	low	2	2	0	0
8	45	low	5	1	0	1

5. CONCLUSIONS AND FUTURE WORK

This work presented HMEDASYS, a medication recognition system integrating an alert system that can be configured to attend to the medication intake of any patient. In the tested scenarios, failures occurred due to the webcam angle of 45 degrees and the low light conditions. These elements were not considered in the creation of the Dataset for the model training. In future work, we plan to enrich the training dataset to deal with more severe imaging conditions. Also, this problem can be reduced by using redundant sensor information, for instance, additional cameras. In future work, we are planning additional pretests using a Raspberry Pi as an edge device, real patients, and better models. Also, integration with Google calendar is an important identified feature as well as the generation of periodic reports for caregivers.

ACKNOWLEDGEMENT

This work was sponsored by Fundação Ensino e Cultura Fernando Pessoa (FECFP) and supported by Artificial Intelligence and Computer Science Laboratory, LIACC.

REFERENCES

Aldeer, M., Javanmard, M., & Martin, R. P. (2018). A review of medication adherence monitoring technologies. Applied System Innovation, 1(2), 14.

Bear, D. M., & Jain, Y. (2011). Medication dispenser with integrated monitoring system. Google Patents.

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection.

Bradski, G., & Kaehler, A. (2008). Learning OpenCV: Computer vision with the OpenCV library. "O'Reilly Media, Inc."

Dazlee, N. M. A. A., Khalil, S. A., Abdul-Rahman, S., & Mutalib, S. (2022). Object Detection for Autonomous Vehicles with Sensor-based Technology Using YOLO. International Journal of Intelligent Systems and Applications in Engineering, 10(1), 129–134.

Faisal, S., Ivo, J., & Patel, T. (2021). A review of features and characteristics of smart medication adherence products. Canadian Pharmacists Journal / Revue Des Pharmaciens Du Canada, 154(5), 312–323. https://doi.org/10.1177/17151635211034198

Forsyth, D. A., & Ponce, J. (2003). A modern approach. In Computer vision: a modern approach. Prentice-Hall.

Jiang, Z., Zhao, L., Li, S., & Jia, Y. (2020). Real-time object detection method based on improved YOLOv4-tiny. ArXiv Preprint ArXiv:2011.04244.

Kaehler, A., & Bradski, G. (2016). Learning OpenCV 3: computer vision in C++ with the OpenCV library. "O'Reilly Media, Inc."

Lam, W. Y., & Fresco, P. (2015). Medication adherence measures: an overview. BioMed Research International, 2015.

McDonnell, P. J., & Jacobs, M. R. (2002). Hospital admissions resulting from preventable adverse drug reactions. Annals of Pharmacotherapy, 36(9), 1331–1336.

Mrosek, R., Dehling, T., & Sunyaev, A. (2015). Taxonomy of health IT and medication adherence. Health Policy and Technology, 4(3), 215–224.

Osterberg, L., & Blaschke, T. (2005). Adherence to medication. New England Journal of Medicine, 353.

Poon, E. G., Keohane, C. A., Yoon, C. S., Ditmore, M., Bane, A., Levtzion-Korach, O., Moniz, T., Rothschild, J. M., Kachalia, A. B., Hayes, J., & others. (2010). Effect of bar-code technology on the safety of medication administration. New England Journal of Medicine, 362(18), 1698–1707.

Pop, C. M., Mogan, G.-L., & Boboc, R. G. (2017). Real-Time Object Detection and Recognition System Using OpenCV via SURF Algorithm in Emgu CV for Robotic Handling in Libraries. International Journal of Modeling and Optimization, 7(5).

Raspberry Pi Foundation. (2022). Raspberry Pi 4 Model B specifications.

Redmon, J. (2013). Darknet: Open Source Neural Networks in C.

Szeliski, R. (2022). Computer vision: algorithms and applications, Second Edition. Springer Cham. https://doi.org/10.1007/978-3-030-34372-9

Tsai, K.-L., Liau, B.-Y., Hung, Y.-M., Yu, G.-J., & Wang, Y.-C. (2020). Development of Smart Pillbox Using 3D Printing Technology and Convolutional Neural Network Image Recognition. Sensors and Materials, 32(5), 1907–1912.

Tzutalin. (2015). LabelImg: graphical image annotation tool for labeling object bounding boxes in images. In GitHub. Tzutalin.

Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., & Yeh, I.-H. (2020). CSPNet: A new backbone that can enhance the learning capability of CNN. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 390–391.

Wehba, S. R., Rinda, J. E., Trohimovich, B. M., & Pelletier, J. (2014). Medication administration and management system and method. Google Patents.

Zou, Z., Shi, Z., Guo, Y., & Ye, J. (2019). Object Detection in 20 Years: A Survey. arXiv. https://doi.org/10.48550/ARXIV.1905.05055