ICCEPM 2022
The 9th International Conference on Construction Engineering and Project Management
Jun. 20-23, 2022, Las Vegas, NV, USA

A Worker-Driven Approach for Opening Detection by Integrating Computer Vision and Built-in Inertia Sensors on Embedded Devices

Sharjeel Anjum[1*], Muhammad Sibtain[2], Rabia Khalid[3], Muhammad Khan[4], Doyeop Lee[5], and Chansik Park [6]

[1,2,3,5,6] *Building Construction Engineering Management, Chung-Ang University, Seoul 06974, South Korea,* E-mail address: sharjeelanjum@cau.ac.kr, sunnyabbas50999@gmail.com, rabiakhalid@cau.ac.kr, doyeop@cau.ac.kr, cpark@cau.ac.kr
[4] *Construction Innovation Integration Laboratory (CII-Lab), Civil, Construction and Environmental Engineering Department, University of Alabama,* E-mail address: mkhan13@crimson.au.edu

Abstract: Due to the dense and complicated working environment, the construction industry is susceptible to many accidents. Worker's fall is a severe problem at the construction site, including falling into holes or openings because of the inadequate coverings as per the safety rules. During the construction or demolition of a building, openings and holes are formed in the floors and roofs. Many workers neglect to cover openings for ease of work while being aware of the risks of holes, openings, and gaps at heights. However, there are safety rules for worker safety; the holes and openings must be covered to prevent falls. The safety inspector typically examines it by visiting the construction site, which is time-consuming and requires safety manager efforts. Therefore, this study presented a worker-driven approach (the worker is involved in the reporting process) to facilitate safety managers by developing integrated computer vision and inertia sensors-based mobile applications to identify openings. The TensorFlow framework is used to design Convolutional Neural Network (CNN); the designed CNN is trained on a custom dataset for binary class openings and covered and deployed on an android smartphone. When an application captures an image, the device also extracts the accelerometer values to determine the inclination in parallel with the classification task of the device to predict the final output as floor (openings/ covered), wall (openings/covered), and roof (openings / covered). The proposed worker-driven approach will be extended with other case scenarios at the construction site.

Key words: Edge Computing, Computer Vision, Inertial Sensors, Safety Monitoring

1. INTRODUCTION

A dynamic and complex working environment and contravening safety rules can expose workers to risk. Due to that, the construction industry has a leading accidents rate than other industries [1]. The most common construction accidents include Falling from Height (FFH), electrocution, being stuck in machinery, and being hit by an obstruction. Among these accidents, the FFH is the most frequent cause of accidents at the construction site [2]. Huang et al. [3] and Kang et al. [4] analyzed the records of Occupational Safety and Health Administration (OSHA) accidents and determined that the FFH raised by 8.3% from 2003 to 2017, which shows that the FFH is a severe problem, including the fall into opening or hole at the temporary supporting platforms such as on-ground openings and scaffolding. There are various safety standards outlined by the (OSHA) such as covering gaps, holes, and openings to prevent worker falls [5].

Two strategies utilized in the construction industry to prevent the injuries and fatalities of the workers are known as proactive and reactive [6]. The proactive method includes training construction workers before working at the construction site, such as the short-term training program. Secondly, the reactive strategy is based on the accident data analysis to identify factors contributing to fatalities on the construction site [2]. The reactive approach is considered more effective due to it's ability to collect data in real-time using sensors and cameras [1]. FFH preventative methods include fixed safety equipment (such as opening covers and guardrails), travel restraint systems (such as safety belts), and fall arrest systems (such as full-body harness). On the other hand, imposing safety rules may improve safety protection equipment and is considered a reactive method to address the construction worker safety issue. The safety manager visits the construction site with the printed checklist for the safety rules compliance, which is a manual approach and is time-consuming and ineffective. Therefore, many researchers utilized the leading technologies such as computer vision and sensors to automate construction workers' safety.

The first leading technology is vision-based, adopted by researchers for worker safety monitoring [7] and worker action recognition for the automated safety inspection process at the construction site [4,8]. Recent research in computer vision-based construction safety monitoring has focused on developing a simple inspection system to detect safety preventive measures. Secondly, researchers utilized the inertia sensors (such as gyroscopes and accelerometers) due to the small size, portability, and low cost for safety monitoring [9,10]. Inertia sensors could be applied in the construction industry to monitor productivity, worker health, and safety [9]. Inertia sensors measure the sudden changes in the acceleration of a body and convert them to electrical signals. It can be used with an accelerometer to monitor position and orientation, but it's more commonly used with a gyroscope (an instrument for measuring angular velocity) to detect orientation changes.

This study proposed a worker-driven approach (a construction worker will report to the safety manager) that extends our prior research work [11]. The main aim of this work is to overcome the technical limitation of previously developed software that can recognize vertical openings (for example, windows "safe") as an on-ground opening (unsafe) which is an erroneous classification. This study integrated built-in inertia sensors with computer vision on an android smartphone to recognize wall opening/covered, floor opening/ covered, and roof opening/covered. This approach is robust, precise, and provides accurate information on the inclination of captured images for the final prediction for the appropriate classification. This worker-driven approach can facilitate the safety management process and keeping records by involving workers.

2. Proposed Method

This research has integrated deep learning-based computer vision with built-in inertia sensors in a smartphone. The paper presents an approach for image recognition and classification of objects by integrating CNN architectures with existing mobile sensing technologies. Figure 1 demonstrates the process required to develop an integrated vision and inertia sensors-based application. We can visualize that the first step is to prepare an image dataset for two categories as opening and covered; once the dataset is prepared, the next step is to set up an anaconda environment for the TensorFlow framework. In the CNN development and training stage, the CNN is designed for custom class classification, training, and inference on the test dataset using TensorFlow [12].

Moreover, deploying the image classification model on edge devices, the TensorFlow model is converted into TensorFlow Lite (Tflite) and attached metadata to specify the model description, such as the input image size and normalization, etc. The converted model and labels file (opening and covered in this scenario) were imported into Android Studio in the last stage. Once a worker captures an image, the application performs two main tasks: (1) it passes the image through the CNN for classification, and (2) it extracts built-in accelerometer sensor values (X, Y, Z-Axis) to determine the inclination. Finally, these extracted values are integrated with the CNN output for a final prediction as floor opening/covered, wall opening/covered, and roof opening/covered, as shown in Figure 2. Finally, the result is uploaded to the real-time database to maintain a record.

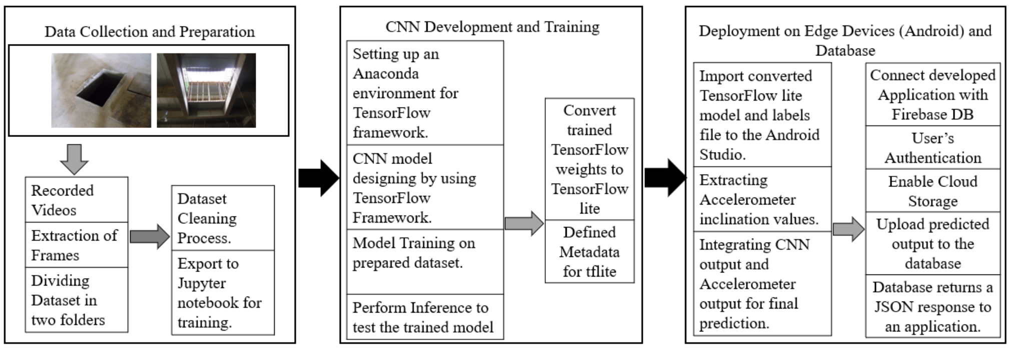

Figure 1. Process Flow of the proposed method

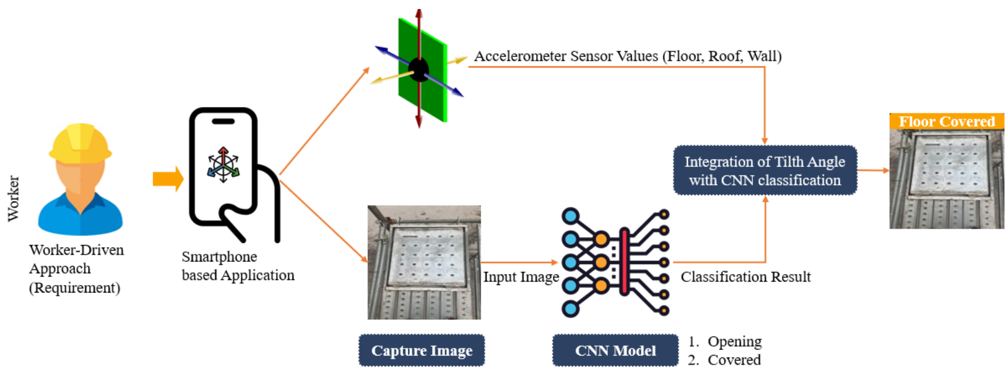

Figure 2. The graphical representation of the proposed worker-driven approach. The application extracts sensor readings parallel with CNN classification and then displays the integrated result at the end.

2.1. Dataset Preparation

To train a deep learning-based image classification model, a large digital image dataset is required. As vision intelligence is an emerging technology in construction, acquiring labeled image datasets remains challenging in this domain. Therefore, the image dataset was obtained from 2 sources (1) Google search engine and (2) recorded multiple videos at Construction Technology and Innovation Laboratory (ConTil), Chung-Ang University, Seoul, South Korea. Random frames were extracted from the recorded videos to prepare the image dataset using the Fast Forward MPEG (Ffmpeg) command-line tool in the Windows operating system. The next step in the dataset preparation is dataset cleaning, which is used to create a valuable dataset for image classification model training and remove unclear/indecent images. It is essential for model training because noise and ineffective image datasets can cause the model's overfitting (learning noise). Moreover, we applied the hold-out technique to avoid overfitting; according to the hold-out approach, a total of

1000 images were prepared for the class "opening" and "covered" and divided into 800, 200 for training and validation, respectively [13].

2.2. Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is utilized in this research work for image classification. The CNN model is designed by stacking multiple convolutional and max-pooling layers for the feature extraction from an input image. The CNN model is trained by providing many relevant and normalized image datasets, which go through layer by layer in a feed-forward manner to classify an image by comparing it with the ground truth. The learning rate of the CNN model was 0.001, the epochs were set to 250, and the adam optimizer was used due to the best experimental performance over all the other optimizers. The error is calculated at the last output layer based on the predicted and actual output differences. Then the backpropagation is performed to overcome the error, and this process continues until the model achieves less error, such as training and validation loss. This study has utilized the TensorFlow library to create an image classification model, an open-source software library used for machine learning algorithms. The developed model can be deployed on various systems using TensorFlow, from edge devices such as mobile and tablets to large computing systems.

Figure 3 shows our image classification-based model, including multiple convolutional and max-pooling layers. The convolutional (Conv2D) layer includes three essential parameters (1) number of filters, (2) kernel size, and (3) activation functions. The first Conv2D layer has 32 filters, with the 3x3 kernel size and Rectified Linear Unit (ReLu) activation function. As the network goes deeper, the number of filters increases with the power of 2. In the remaining Conv2D layers, the number of filters was assigned as 64, 128, and 256 for the 2nd, 3rd, and 4th, respectively. Although, the kernel size and activation function are the same for all layers. The next layer is max-pooling, which reduces the number of feature maps and amount of computing performed in a network; it is stacked next to each Conv2D layer. Following thorough, the next step is the using flatten layers; the output of the convolutional layers is in the form of 2 dimensions (2D), but the fully connected layers require 1D data; therefore, the flatten layer is attached before passing data to the fully connected layers (Dense layers). At the end of the network, we utilized two fully connected layers, which take high-level filtered images and convert them to a vector and a sigmoid activation function for binary classification (Covered or Opening) [12]. This CNN model used a one-node technique at the output layer, owing to the advantage of requiring few weights and biases.

2.3. Deployment on Edge Devices

This section explains a process to convert and deploy the trained model from TensorFlow to TensorFlow lite (TFLite) to perform inference on edge devices. TFLite comprises tools for machine learning-related tasks and visualizing inference on edge devices such as smartphones and IoT to evade server-client round-trips without requiring the internet [14].

Two main pre-processing steps are required to deploy the Deep Learning model on edge devices:

- The TFLite converter converts the trained model into TFLite format that requires the TensorFlow model as input and generates the TfLite model (an optimized Flat Buffer format identified by .tflite file extension). Additionally, the quantization technique is applied during conversion to limit the model size to 250 MB as per the requirement of the Android Studio development environment.
- Another essential step is to generate metadata that describes model information such as input information, normalization of input data, output information, and labels. The normalization technique is utilized to convert values to a common scale. The authors set

the normalization value 127.5 for the model mean and standard deviation. Finally, the quantized model is exported into Android Studio and used to develop the proposed integrated application.

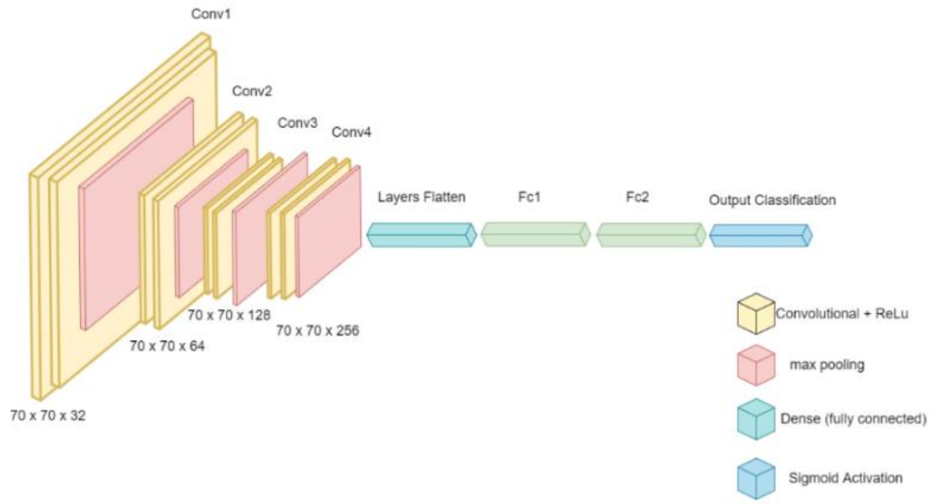

Figure 3. Designed CNN for binary image classification

2.4. Built-In Inertia Sensors

Smartphones and tablets have become a significant part of our daily life. Smartphones, in particular, need to be one-time calibration to ensure that they provide accurate readings for both motion and orientation data. Inertia sensors are often used in robotics to detect a moving object's orientation and velocity. Most commercially available smartphones include an accelerometer, gyroscope, ambient temperature sensor, light sensor, barometer, proximity sensor, and GPS, among other sensing technologies. The device's acceleration is measured via accelerometer sensors; all three axes of the accelerometer, X, Y, and Z, can be read. The gyroscope detects the smartphone's roll, pitch, and yaw motions about the X, Y, and Z axes, which detects the device's rotation rate. In this study, we need the inclination angle of the smartphone to integrate with the CNN output for final classification. We have cloned the available open-source code of the BasicAirData Clinometer application. This simple android-based application uses onboard accelerometers to measure the inclination angles of the device regarding gravity's direction [15]. So, we have manually analyzed and extracted all three axis values on different angles with different ranges. Based on these values, an application predicts the final output as horizontal (floor) opening/covered, roof opening/covered, and vertical (wall) opening/covered. The application required a one-time calibration (to overcome uncertainty) of the built-in sensors in a smartphone to give the best performance.

The extracted accelerometer values used to predict the final output can be seen in Table 1. Based on the extracted values, an application determines the device's inclination. If the X-Axis value ranges between -65 to 30, Y-Axis -10 to 50, and Z-Axis -30 to 30 while capturing an image, this CNN output integrated with an accelerometer sensor is classified as a horizontal inclination (Floor opening / covered). Suppose the X-Axis value ranges between -30 to 30, Y-Axis 50 to 90, and Z-Axis -40 to 40 while capturing an image. In that case, this CNN output is integrated with an accelerometer sensor classified as a vertical inclination (Window opening / covered). Finally, suppose the Z-Axis is 40 to 90 while capturing an image. In that case, this CNN output is integrated with an accelerometer sensor classified as roof opening / covered.

Table 1. Accelerometer Values for Inclination

X-Axis	Y-Axis	Z-Axis	Output (Inclination)
-65 to 30	-10 to 50	-30 to 30	Horizontal (Floor)
-30 to 30	50 to 90	-40 to 40	Vertical (Window)
-	-	40 to 90	Roof (Scaffolding)

3. Results and Evaluation

The developed application is tested at the ConTIL. Figure 4 depicts the graphical user interface and the final prediction by fusing built-in inertia sensors with computer vision. We have tested an application for a horizontal opening, horizontal covered, roof opening, and roof covered. Note that, due to the limited access to the actual construction site, we assumed scaffolding opening/covered with a horizontal inclination of accelerometer sensor as floor opening/covered, and scaffolding opening/covered image captured from the first floor of scaffolding as roof opening/covered. The trained model was evaluated using a confusion matrix, where TP (opening) was 127, FP was 5, TN (covered) was 65, and FN was 3, which can be seen in Figure 5. Moreover, Table 2 shows the CNN model's evaluation matrices, which achieved precision, recall, F1-score, and validation accuracy with 96.9%, 97.6%, 97.24%, and 96.5%, respectively, with a validation loss of 0.017, which is commendable.

4. Conclusion and Future Work

This research proposed a worker-driven approach for opening detection using vision-based and built-in inertia sensors-based integrated android applications to classify different openings at the construction site. The designed CNN is trained on two classes of opening and covered custom dataset based on the CNN output; the developed application extracts accelerometer values to determine the inclination of the device to predict final output as floor opening/covered, wall opening/covered, and roof opening/covered. Following that, the integration of computer vision and inertia sensors is performed. The final predicted output is uploaded into the real-time firebase database to keep track of the inspection. This approach helps prevent falls by involving workers in the reporting process for safety rules compliance. This developed application would be extended with other safety rules compliance for the lenience of the safety inspector. To promote the worker-driven approach, incentives in coins or an appreciation system are required. Consequently, the proposed method can be improved by incorporating an appropriate reward mechanism.

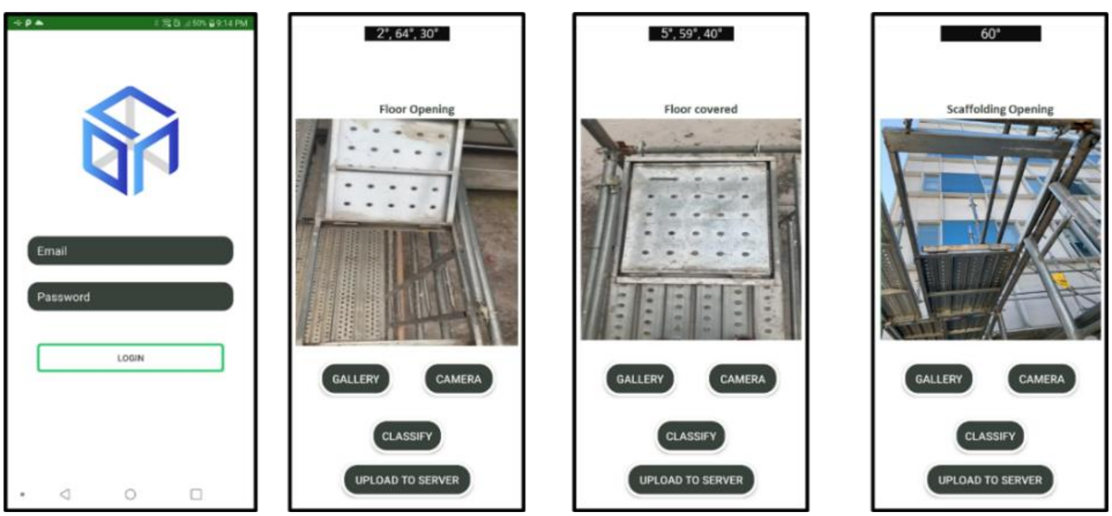

Figure 4. The user interface of the developed application

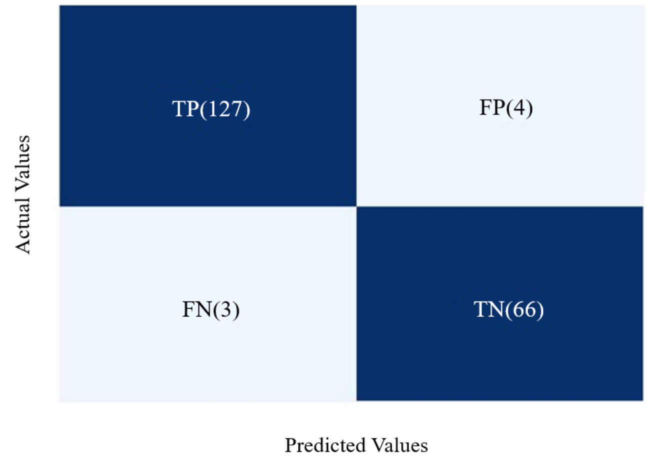

Figure 5. Confusion Matrix for binary classification

Table 2. Evaluation Matrices for CNN

Evaluation Index	**Test Results**
Precision	96.9%
Recall	97.6%
F1-Score	97.24%
Validation Loss	0.017
Accuracy	96.5%

ACKNOWLEDGMENTS

This research was conducted with the support of the “National R&D Project for Smart Construction Technology (No.22SMIP-A158708-03),” funded by the Korea Agency for Infrastructure Technology Advancement under the Ministry of Land, Infrastructure, and Transport, and managed by the Korea Expressway Corporation and supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2020R1A4A4078916).

REFERENCES

[1] S. Anjum, N. Khan, R. Khalid, M. Khan, D. Lee, C. Park, Fall Prevention From Ladders Utilizing a Deep Learning-Based Height Assessment Method, IEEE Access. 10 (2022) 36725–36742. https://doi.org/10.1109/ACCESS.2022.3164676.

[2] E.A. Nadhim, C. Hon, B. Xia, I. Stewart, D. Fang, Falls from height in the construction industry: A critical review of the scientific literature, Int. J. Environ. Res. Public Health. 13 (2016) 638. https://doi.org/10.3390/ijerph13070638.

[3] X. Huang, J. Hinze, Analysis of Construction Worker Fall Accidents, J. Constr. Eng. Manag. 129 (2003) 262–271. https://doi.org/10.1061/(asce)0733-9364(2003)129:3(262).

[4] Y. Kang, S. Siddiqui, S.J. Suk, S. Chi, C. Kim, Trends of Fall Accidents in the U.S. Construction Industry, J. Constr. Eng. Manag. 143 (2017) 04017043. https://doi.org/10.1061/(asce)co.1943-7862.0001332.

[5] OSHA, Clarification of fall protection requirements for open holes on a construction site. | Occupational Safety and Health Administration, n.d. https://www.osha.gov/laws-regs/standardinterpretations/1998-11-17 (accessed February 3, 2022).

[6] W. Xu, T.K. Wang, Dynamic safety prewarning mechanism of human–machine–environment using computer vision, Eng. Constr. Archit. Manag. 27 (2020) 1813–1833.

https://doi.org/10.1108/ECAM-12-2019-0732.
[7] W. Fang, B. Zhong, N. Zhao, P.E.D. Love, H. Luo, J. Xue, S. Xu, A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network, Adv. Eng. Informatics. 39 (2019) 170–177. https://doi.org/10.1016/J.AEI.2018.12.005.
[8] M. Khan, R. Khalid, S. Anjum, N. Khan, S. Cho, C. Park, Automation in Construction Tag and IoT based safety hook monitoring for prevention of falls from height, Autom. Constr. 136 (2022) 104153. https://doi.org/10.1016/j.autcon.2022.104153.
[9] S.J. Ray, J. Teizer, Real-time construction worker posture analysis for ergonomics training, Adv. Eng. Informatics. 26 (2012) 439–455. https://doi.org/10.1016/j.aei.2012.02.011.
[10] M. Khan, R. Khalid, S. Anjum, N. Khan, C. Park, IMU based Smart Safety Hook for Fall Prevention at Construction Sites, TENSYMP 2021 - 2021 IEEE Reg. 10 Symp. (2021) 1–6. https://doi.org/10.1109/TENSYMP52854.2021.9550944.
[11] S. Anjum, R. Khalid, M. Khan, N. Khan, C. Park, A Pull-Reporting Approach for Floor Opening Detection Using Deep-Learning on Embedded Devices, in: Proc. 38th Int. Symp. Autom. Robot. Constr., International Association for Automation and Robotics in Construction (IAARC), 2021. https://doi.org/10.22260/isarc2021/0055.
[12] R. Yamashita, M. Nishio, R. Kinh, G. Do, K. Togashi, Convolutional neural networks: an overview and application in radiology, (n.d.). https://doi.org/10.1007/s13244-018-0639-9.
[13] 8 Simple Techniques to Prevent Overfitting | by David Chuan-En Lin | Towards Data Science, (n.d.). https://towardsdatascience.com/8-simple-techniques-to-prevent-overfitting-4d443da2ef7d (accessed September 28, 2021).
[14] Jim Su, How to Train a Custom TensorFlow Lite Object Detection Model, (2020).
[15] GitHub - BasicAirData/Clinometer: A Clinometer for Android mobile devices, (n.d.). https://github.com/BasicAirData/Clinometer (accessed December 30, 2021).