Deep-Learning and Stereo Camera Based System for Real-Time Virtual Safety Boundaries

Juraj Repka, Juraj Ďuďák
Slovak University of Technology
Faculty of Materials Science and Technology in Trnava
Jána Bottu 25, Trnava, Slovakia
{juraj.repka, juraj.dudak}@stuba.sk

Abstract. *We propose a real-time, cost-effective system for implementing a virtual safety zone that enables monitoring of a person's position relative to user-defined virtual boundaries. Our approach employs a hybrid algorithm that combines depth mapping and disparity calculation using deep learning models applied to stereo image pairs. Using a stereo camera setup, the system estimates human keypoint positions, which are then compared against the predefined zones. When a zone is violated, a corresponding trigger is generated. Although the system has shown its effectiveness, it also faces challenges in achieving high accuracy, making it less suitable for applications requiring precise localization.*
Keywords. Virtual protective zone, deep learning, stereo vision, human tracking.

1 Introduction

The rise of automation in industrial environments has significantly improved production efficiency, product quality, and workplace safety. Machines can now operate 24/7, reduce human error, and perform tasks that may be too dangerous for workers. Despite these advantages, automation presents serious challenges, particularly in ensuring safe interaction between humans and machines (Rybski et al., 2012).

Many industrial systems still rely on older technologies that lack contextual awareness and dynamic safety mechanisms. This means that even a minor mistake, such as an operator unintentionally entering a hazardous zone, can lead to serious accidents if machines are not equipped to detect such situations. Although newer industrial equipment may include built-in safety logic, a large portion of existing systems continues to rely on manual safety protocols (Mohammadi Amin et al., 2020; Rybski et al., 2012).

This paper is motivated by the need for a flexible, intelligent safety system that can automatically monitor safety zones and identify when a person violates them. The primary goal is to design and implement a virtual safety zone system that uses 3D computer vision and deep learning techniques to detect people in real time and assess their position relative to dangerous areas. Compared to traditional methods such as light curtains or infrared sensors, 3D vision allows for greater adaptability and accuracy, especially in complex environments (Mosberger et al., 2014; Zhou et al., 2022).

In this work, we created a working prototype capable of detecting individuals in a monitored 3D space and determining whether they are within a customizable danger zone. The system is also designed to provide visual feedback and alerts in real time to the operator.

We are utilizing stereo vision and deep learning models to create a cost-effective system for human position monitoring. We developed our own stereo system but also used an OAK-D-PRO commercial one. We applied multiple depth estimation techniques and also signal processing techniques such as smoothing algorithms.

2 Related Work

Recently, there has been a growing interest in improving safety within industrial environments using advanced technologies such as computer vision and sensor-based systems. Traditional solutions, such as light curtains or mechanical interlocks, often lack the flexibility and context awareness required to adapt to dynamic workspaces. As a result, research has increasingly focused on intelligent systems that can respond in real-time to environmental changes and reduce the risk of human error.

One significant approach involves the use of 3D cameras to detect human presence near hazardous machinery. Cheng Zhou and his colleagues developed a system that monitors the area surrounding robotic arms using 3D visual data (Zhou et al., 2022). These systems enable cameras to be placed at a greater distance compared to traditional light barriers, which must be positioned close to the machine. This configuration reduces false positives caused by non-human objects and improves overall reliability.

Sotiris Makris and his team introduced a more ad-

vanced concept, which proposed dynamic safety zones instead of static ones (Makris, 2021). These zones adjust in real-time on the basis of operational requirements and human-robot interaction. Such an approach allows for closer, yet safe collaboration between humans and machines, thereby improving productivity without compromising safety.

For mobile robotic platforms that navigate freely throughout the facility, the safety challenge becomes even more complex. Hyunjoong Cho et al. addressed this problem by developing a system that detects objects of interest and generates protective zones around them based on contextual factors, such as robot speed (Cho et al., 2022). The faster the robot moves, the larger the protective zone becomes, providing a dynamic buffer that minimizes the risks of collision.

Juraj Slovák et al. proposed a hybrid vision and RTLS-based safety system that adapts robot behavior according to the proximity and identity of nearby objects, including humans (Slovák et al., 2021). Their approach combines depth camera data with real-time location systems to distinguish between authorized (e.g., supply trolleys) and unauthorized (e.g., human) entries into safety zones, reducing unnecessary downtimes without compromising safety.

Additionally, Kozamernik et al. introduced a visual quality and safety monitoring system for human–robot collaboration that combines stereo and depth cameras with deep learning-based object and posture recognition (Kozamernik et al., 2023). Their system supports both safety (e.g., hand detection and posture analysis) and quality assurance (e.g., inspection of final assemblies), demonstrating that safety and performance feedback can be achieved with minimal hardware overhead.

All of these approaches share a common foundation in leveraging 3D computer vision to detect humans and define intelligent safety mechanisms. They demonstrate a shift toward autonomous systems capable of adapting to varied and evolving industrial environments. These insights have informed the development of our proposed solution, which aims to detect breaches of safety zones in a cost-effective and highly adaptable manner suitable for wide deployment.

3 System Architecture

The proposed virtual security zone system is made up of three main modules: a stereo vision module, a processing module, and a visualization module. These components operate in real time to detect human presence, estimate 3D position using stereo cameras and deep learning, and determine whether individuals are within a predefined virtual safety zone. The system includes hardware-based depth estimation, keypoint detection models, and a web-based visualization interface for interactive zone management (see Fig. 1).

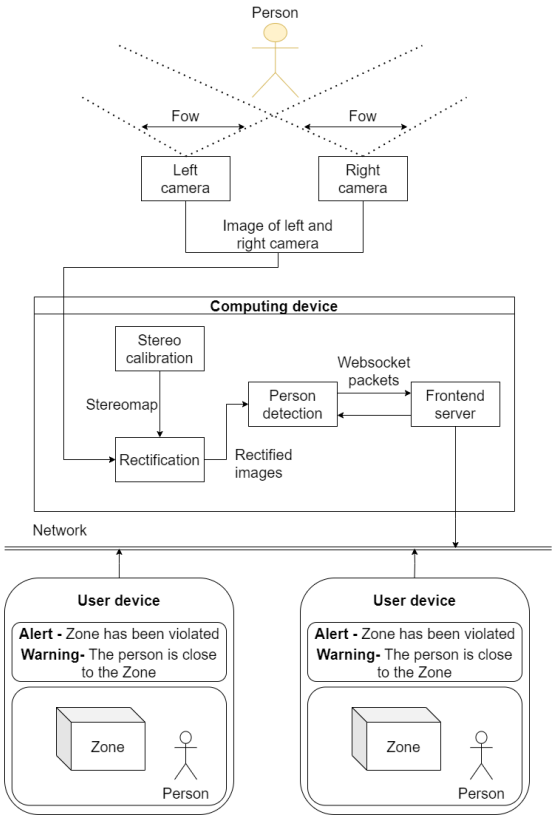

Figure 1. System architecture diagram

3.1 Stereo Vision Module

The stereo vision module is responsible for capturing the real-world 3D scene and generating a depth map. Using stereo vision technology, it emulates human depth perception by using two spatially separated cameras (see Fig. 2). These cameras are offset by a fixed baseline, a critical parameter that influences depth accuracy. A wider baseline improves depth estimation for distant objects by increasing disparity, whereas a narrower baseline is preferable for nearby objects to avoid excessive disparity and reduce matching errors.

Depth estimation is performed via triangulation, a geometric method that calculates the 3D position of a point based on its disparity in the horizontal shift between corresponding points in the left and right images. Larger disparities indicate closer objects; smaller disparities suggest objects are farther away. The accuracy of the depth calculation depends on factors such as the baseline length, image resolution, and stereo calibration quality.

For effective stereo vision, the camera pair must be calibrated and the images corrected so that the corresponding points lie on the same horizontal scan lines (epipolar lines). This rectification step is essential for robust disparity computation and is a fundamental part of our stereo vision software implementation.

We used two different stereo vision camera implementations in our research. The first is our custom-built system (see Fig. 3), also used in our previous

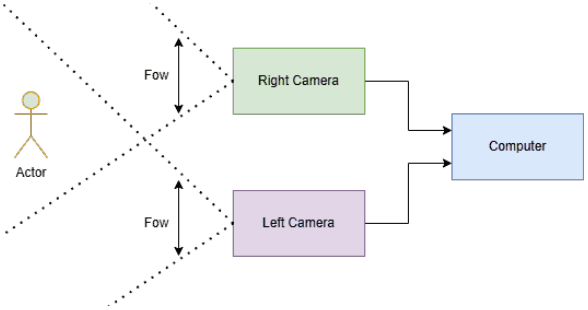

Figure 2. Abstraction of stereovision

work (Sládek et al., 2024). This setup consists of two standard webcams mounted on a rigid fixed baseline. Fixated positioning is essential to maintain consistent calibration, which is handled by our software. This calibration process estimates both the intrinsic and the extrinsic parameters of the stereo camera system, which are crucial for accurate depth estimation.

Figure 3. Homemade stereo vision camera

The second implementation uses the commercially available OAK-D Pro camera developed by Luxonis (see Fig. 4). This device integrates both stereo vision sensors and an onboard processor, enabling on-device depth estimation and neural inference. This hardware acceleration allows us to offload computationally intensive tasks to the device, significantly improving overall processing speed.

Figure 4. OAK-D-PRO camera

All algorithms were implemented and tested using both stereo vision systems. However, in this paper, we focus exclusively on the implementation using the OAK-D Pro. This decision is based on the superior performance of the OAK-D Pro, which benefits from its own hardware-accelerated stereo matching pipeline and optimized internal processing. In addition, the advanced camera specifications contributed to its improved results. For these reasons, we selected the OAK-D Pro as the primary stereo vision device for our system.

3.2 Processing Module

The processing module is the next component in our system's data flow. It receives images from the stereo camera, along with the necessary camera parameters for depth calculation. This module is responsible for detecting person keypoints, computing depth information, and forwarding the processed data to the Visualization module.

The processing module incorporates two independent processing algorithms that operate on stereo images acquired from a stereo vision module.

The two implemented algorithms are as follows: (i) a depth map-based algorithm and (ii) a disparity-based algorithm utilizing keypoint detections from stereo image pairs. Both algorithms are designed to function independently and work differently within the processing pipeline.

Depth Map-based Algorithm

This algorithm operates directly on the real-time depth map computed by the stereo device's onboard processor. It follows the depth-lookup strategy demonstrated in Luxonis' OAK-D calc-spatials-on-host reference implementation (Luxonis Inc., 2023). The primary steps are:

1. The stereo camera computes a depth map directly on the device using its onboard processor.
2. A deep learning model, specifically MediaPipe Pose (Bazarevsky et al., 2020), is applied to detect the human subject and extract keypoint coordinates from a single image frame.
3. For each detected keypoint, the corresponding depth value is retrieved from the depth map using its pixel coordinates.

The depth map used in this algorithm is calculated by the internal OAK-D-Pro processor, allowing faster performance compared to a disparity-based algorithm.

Disparity-based Algorithm

We developed a disparity-based algorithm that runs entirely on the system's main processing unit. This algorithm performs keypoint detection independently on

both the left and right stereo images. The main steps of our method are as follows:

1. Two independent instances of the MediaPipe Pose model are applied, one for the left image and one for the right image, due to the model's internal state management, which improves detection stability but prevents reusability across frames.
2. The key points are detected independently in both images.
3. For each pair of corresponding key points, the disparity is calculated as the horizontal difference of pixels between their positions in the left and right images (see Fig. 5).
4. The depth is then estimated on the basis of the computed disparity using stereo vision geometry.

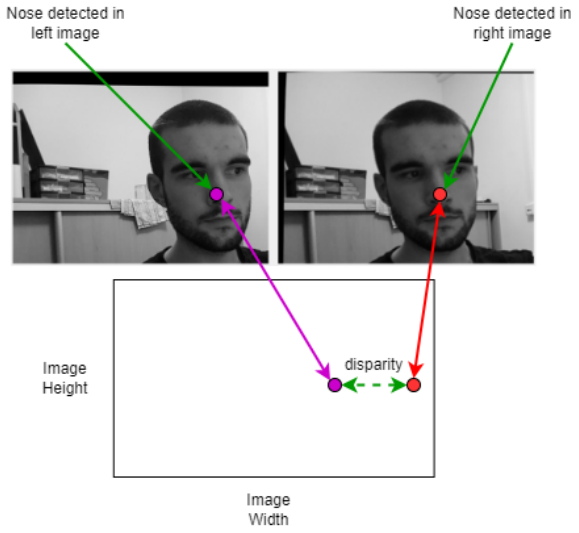

Figure 5. Disparity visualization

Although this approach requires running two models in parallel, the computational load remains manageable, as MediaPipe Pose is optimized for low-latency performance and can be executed efficiently even on edge devices.

3.3 Visualization Module

The Visualization module is responsible for displaying the detected 3D keypoints received from the processing module, as well as the boundaries of user-defined zones. It also allows users to define and modify zone parameters interactively.

Implemented as a web-based application, the module communicates with the processing module via WebSockets to receive real-time data. Display 3D keypoints using React for front-end development and Three.js for 3D rendering (see Fig. 6).

In addition to visualization, the module includes logic to determine whether detected keypoints fall

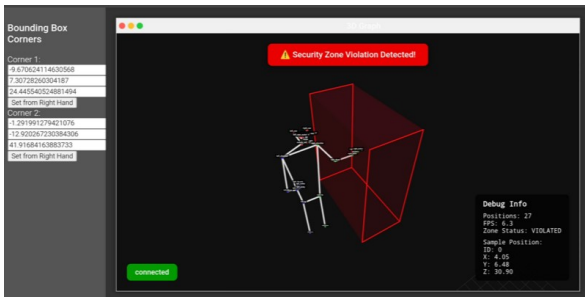

Figure 6. Frontend visualization

within predefined zones. If a person enters a defined zone, an on-screen alarm is triggered. The alarm is deactivated once the person exits the zone.

4 Experiments

Stereo vision systems often face two primary challenges that can degrade performance: poor lighting conditions and low image contrast.

To evaluate the robustness of our algorithms under these conditions, we conducted four experiments using two distinct environments: a residential room and an office room. These settings were selected because of their differing background contrasts; the residential room featured higher contrast compared to the office environment.

The experiments were conducted as follows:

- Low-light residential room.
- Normal-light residential room.
- Low-light office room.
- Normal-light office room.

Each algorithm was tested in all four scenarios, enabling us to isolate and analyze the impact of lighting and contrast on stereo vision performance.

The **Disparity-based Algorithm** exhibited significant performance degradation under low light conditions. Poor lighting adversely affected the accuracy of person detection by deep learning models, which in turn caused higher errors in disparity estimation. These errors manifested as incorrect or noisy depth values. However, this algorithm performed comparably in both residential and office settings, suggesting that it is relatively robust to variations in image contrast.

The **Depth Map-based Algorithm**, on the other hand, was sensitive to both low light and low contrast environments. This method relies on stereo matching, which attempts to find corresponding points between the left and right camera images to compute depth. Low contrast made this matching process more difficult, while low light further reduced the image quality. However, the contrast issue was somewhat mitigated

Table 1. Quantitative performance of the depth-map and disparity pipelines under four illumination/contrast scenarios.

Scenario	Algorithm	MAE (cm)	RMSE (cm)
Normal-light – residential	Depth-map	2.9 ± 0.4	4.3 ± 0.6
	Disparity	3.5 ± 0.5	5.1 ± 0.7
Normal-light – office	Depth-map	4.4 ± 0.5	6.5 ± 0.7
	Disparity	4.1 ± 0.6	6.0 ± 0.8
Low-light – residential	Depth-map	7.0 ± 0.8	10.3 ± 1.2
	Disparity	5.6 ± 0.7	8.2 ± 1.0
Low-light – office	Depth-map	8.6 ± 1.0	12.5 ± 1.4
	Disparity	7.2 ± 0.9	10.9 ± 1.3

by using the OAK-D Pro built-in infrared (IR) projector, which projects an active pattern on the scene. This projection enhances texture in otherwise low-contrast regions, improving stereo correspondence and depth estimation.

We also identified two additional faults that are not related to environmental conditions, but stem from system-level issues.

The first issue was detected in the depth map–based algorithm, which occasionally produced incomplete depth maps. Specifically, some regions within the map contained anomalous values or failed to compute depth altogether (see Fig. 7). These inaccuracies pose a significant problem during depth extraction, as the resulting data is unreliable.

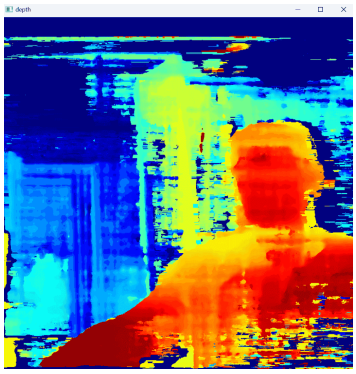

Figure 7. Depth map before postprocessing

To address this issue, we first applied an outlier removal step using local statistics. For each pixel, we computed the mean and standard deviation within a neighborhood $n \times n$ and zeroed any pixel values that significantly deviated from the local mean, based on a predefined threshold. We then used the Telea inpainting algorithm (Telea, 2004) to reconstruct the missing (zero) values by interpolating from nearby valid pixels. Finally, a bilateral filter was applied to smooth the depth map while preserving important edge details (see Fig. 8).

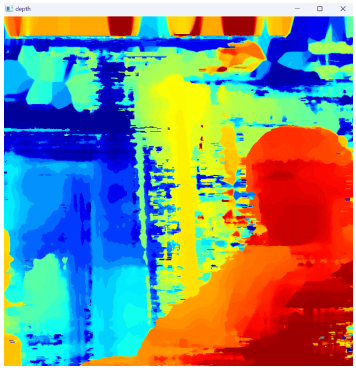

Figure 8. Depth map after postprocessing

These steps improved the overall smoothness and continuity of the depth map. However, they did not fully resolve the underlying issue and, in some cases, introduced new artifacts.

The second fault was observed in the disparity-based algorithm. This issue was traced to the deep learning–based object detectors. Since these detectors do not operate with 100% accuracy, occasional misdetections led to unstable disparity estimations. As a result, the depth calculated would sometimes fluctuate dramatically, jumping to abnormally low or high values, which in turn caused false activations of the safety zone (see Fig. 9).

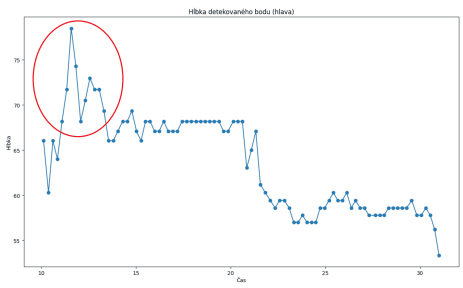

Figure 9. Temporal variation in depth due to detection anomalies

To mitigate this, we experimented with both Kalman filtering and exponential smoothing to suppress sudden peaks and stabilize the depth output. After testing, we found that exponential smoothing yielded better results. Although Kalman filtering effectively reduced noise, it also overly smoothed rapid intentional movements, such as a person walking quickly, causing a loss of responsiveness. In contrast, exponential smoothing maintained stability while adapting more gracefully to fast motion (see Fig. 10).

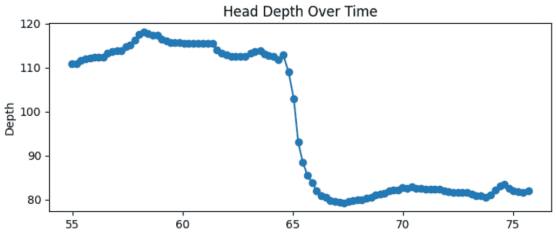

Figure 10. Depth stabilized using exponential smoothing

As before, this approach improved performance but did not completely resolve the problem. Therefore, we implemented a final hybrid solution that combines both depth map-based and disparity-based algorithms, leveraging the strengths of each to enhance robustness and reliability.

5 Hybrid approach

Based on the experiments conducted with our system, we developed a final solution that combines two algorithms into a hybrid approach. This methodology leverages the strengths of each algorithm while mitigating their respective weaknesses. The proposed approach is divided into two main stages. Depth Calculation and Post-processing.

Depth Calculation

Our depth calculation process integrates both algorithms. Initially, we perform person detection independently in both left and right images captured by the stereo camera. This detection yields the corresponding coordinates for keypoints in each image, enabling depth computation through a triangulation based on disparity.

A crucial improvement in our method is the constraint applied during the matching phase: we only accept corresponding points that lie within a 10-pixel threshold of their respective epipolar lines. This constraint significantly reduces the number of erroneous detections.

Subsequently, we employ a depth-map-based algorithm with specific enhancements. Instead of directly using the depth value from a single pixel, we extract a 5×5 pixel neighborhood surrounding the detected keypoint and calculate the average depth within this window. As a result, we obtain two distinct depth measurements:

1. Depth derived from triangulation (d_{tri})
2. Depth derived from the depth map average (d_{map})

The final depth is computed using a weighted combination of these measurements:

$$w_{\text{tri}} = \min\left(1.0, \frac{\text{disparity}}{\text{disparity\_threshold}}\right) \quad (1)$$

$$w_{\text{depth}} = 1.0 - w_{\text{tri}} \quad (2)$$

$$D_{\text{final}} = d_{\text{tri}} \cdot w_{\text{tri}} + d_{\text{map}} \cdot w_{\text{depth}} \quad (3)$$

Where:

- w_{tri}: weight for triangulation depth (increases with disparity)
- w_{depth}: complementary weight from depth map
- d_{tri}: depth from stereo triangulation
- d_{map}: average depth around keypoint in depth map
- D_{final}: final depth estimate

The disparity threshold defines the point at which the system begins relying solely on depth calculated from disparity-based algorithm. In our implementation, we used a threshold value of 30 pixels. However, we did not conduct a systematic evaluation of alternative threshold values, so there may be opportunities for further accuracy improvements through parameter tuning.

This weighting scheme ensures that when a person is located further from the camera (where disparity calculations tend to be more error-prone), we rely more heavily on the depth map-based measurement. In contrast, closer objects benefit from more accurate disparity calculations.

Post-processing

The post-processing stage is designed to further refine the calculated depth and is divided into two steps:

1. **Outlier Removal**: We utilize the Z-score method to identify and discard anomalous depth points, significantly improving the reliability of our measurements.
2. **Smoothing**: Depth coordinates are accumulated within a temporal window. If a stationary state is identified across five consecutive windows, robust smoothing is applied:

   $$\text{position} = 0.98 \cdot \text{last\_stable} + 0.02 \cdot \text{current}$$

   In dynamic scenarios, median filtering followed by exponential smoothing is applied to effectively minimize noise.

Post-processing Hyperparameters. The following hyperparameters are used during the post-processing stage:

- **Z-score threshold:** A threshold of 2.5 is applied to detect outliers.
- **Depth window:** A 5×5 pixel mean filter is computed around each landmark.
- **Epipolar tolerance:** Vertical mismatches are tolerated up to 10 pixels.
- **Temporal median:** A history of $N = 7$ frames is used for temporal median filtering.
- **Stationary lock:** Activated after 5 consecutive "stable" detection windows.
- **Locked smoothing:** When locked, smoothing is applied as $0.98\times$ previous $+0.02\times$ current value.
- **Dynamic smoothing:** An exponential smoothing factor α in the range $[0.80, 0.90]$ is used.

This dual-step smoothing strategy considerably reduces anomalies in the measured depth of human keypoints, thereby minimizing false activations of the safety zone.

6 Conclusion and Future Work

In this work, we addressed the challenge of implementing virtual safety zones using 3D computer vision. Our main objective was to design and develop a system capable of detecting individuals in a monitored physical space and determining whether they have entered a predefined virtual boundary. Upon zone violation, the system generates a real-time alert.

We proposed a stereo vision–based system integrated with deep learning for human detection. Two primary depth estimation approaches were evaluated: a triangulation-based method using independent keypoint detections from stereo image pairs, and a depth map–based method relying on a single-image keypoint detection with depth values retrieved from a stereo generated map. Each approach demonstrated strengths and weaknesses. The triangulation method provided accurate results for nearby objects, but suffered from occasional detection anomalies. The depth map approach was more stable in keypoint detection, but less reliable in poorly lit or low-texture environments, with incomplete depth maps.

To address these limitations, we implemented several post-processing techniques, including outlier removal, image inpainting, and smoothing using Kalman filtering and exponential smoothing. Although these techniques improved overall stability, they also introduced trade-offs such as delayed system response and increased computational load.

Ultimately, we proposed a hybrid approach that combines both triangulation and depth map–based estimates. This solution balances the strengths of each method, producing more accurate and reliable results at the cost of higher computational demands. In parallel, we developed an interactive visualization interface for defining and monitoring virtual safety zones, detecting zone violations, and visualizing detected keypoints.

Despite the system meeting its core objectives, several limitations remain, most notably in depth estimation accuracy and the stability of stereo keypoint correspondence. Therefore, we propose several avenues for future work:

- **Improved Neural Architectures:** Develop or integrate deep learning models that leverage both stereo image inputs while respecting epipolar constraints to improve correspondence matching and keypoint consistency.
- **Multi-camera Integration:** Extend the system to support multiple stereo camera setups for increased spatial coverage and accuracy, especially in larger or occluded environments.
- **Multi-person Tracking:** Implement advanced tracking mechanisms to enable simultaneous monitoring of multiple individuals and reduce false zone violations.

Our findings indicate that combining stereo vision and deep learning offers a viable path toward intelligent safety zone monitoring. However, due to its current limitations, we do not yet recommend this approach for safety-critical environments requiring high-precision tracking, until the proposed enhancements are implemented and validated.

Privacy and ethics

The proposed system is currently in the proof-of-concept stage. In its current version, it is not deployed in an industrial environment. When processing data, the source data is deleted after the camera data has been read. The system does not archive any data that could be attributed to a specific person.

Acknowledgments

This work was supported by KEGA through the Teaching and Development of methodology for the use of microcontrollers in automation using practical examples and laboratory exercises for engineering students, under Grant 024STU-4/2024.

References

Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T. L., Zhang, F., & Grundmann, M. (2020). Blazepose: On-device real-time body pose tracking. *ArXiv*, *abs/2006.10204*. https : / / api . semanticscholar.org/CorpusID:219793039

Cho, H., Lee, K., Choi, N., Kim, S., Lee, J., & Yang, S. (2022). Online safety zone estimation and violation detection for nonstationary objects in workplaces. *IEEE Access*, *10*, 39769–39781. https://doi.org/10.1109/ACCESS.2022.3165821

Kozamernik, N., Zaletelj, J., Kosir, A., Suligoj, F., & Bracun, D. (2023). Visual quality and safety monitoring system for human-robot cooperation. *The International Journal of Advanced Manufacturing Technology*, *128*, 685–701. https://doi.org/https://doi.org/10.21203/rs.3.rs-2409100/v1

Luxonis Inc. (2023). Calculate spatial coordinates on the host [DepthAI Examples repository (sample code and README explain depth-lookup for arbitrary key-points); accessed 28 Jun 2025.].

Makris, S. (2021). Dynamic safety zones in human robot collaboration. In *Cooperating robots for flexible manufacturing* (pp. 271–287). Springer International Publishing. https://doi.org/10.1007/978-3-030-51591-1_14

Mohammadi Amin, F., Rezayati, M., van de Venn, H. W., & Karimpour, H. (2020). A mixed-perception approach for safe human–robot collaboration in industrial automation. *Sensors*, *20*(21). https://www.mdpi.com/1424-8220/20/21/6347

Mosberger, R., Andreasson, H., & Lilienthal, A. J. (2014). A customized vision system for tracking humans wearing reflective safety clothing from industrial vehicles and machinery. *Sensors*, *14*(10), 17952–17980. https://www.mdpi.com/1424-8220/14/10/17952

Rybski, P., Anderson-Sprecher, P., Huber, D., Niessl, C., & Simmons, R. (2012). Sensor fusion for human safety in industrial workcells. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3612–3619. https : / / doi . org / 10 . 1109 / IROS.2012.6386034

Sládek, I., Gašpar, G., Repka, J., & Budjac, R. (2024). Stereo vision based 3d positioning for real-time control applications. *Proceedings of 35th International scientific conference CECIIS*.

Slovák, J., Melicher, M., Šimovec, M., & Vachálek, J. (2021). Vision and rtls safety implementation in an experimental human—robot collaboration scenario. *Sensors*, *21*(7). https : / / www . mdpi . com / 1424-8220/21/7/2419

Telea, A. (2004). An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, *9*(1), 23–34. https://doi.org/10.1080/10867651.2004.10487596

Zhou, C., Ren, D., Zhang, X., Yu, C., & Ju, L. (2022). Human position detection based on depth camera image information in mechanical safety. *Advances in Mathematical Physics*. https://doi.org/10.1155/2022/9170642