Multi-view Real-time 3D Occupancy Map for Machine-patient Collision Avoidance

Timothy Callemein, Kristof Van Beeck and Toon Goedemé
EAVISE, PSI, KU Leuven, Jan Pieter de Nayerlaan 5, Sint-Katelijne-Waver, Belgium

Keywords: Cobots, Real-time, 3D Occupancy, Multi-view.

Abstract: Nowadays - due to advancements in technology - cooperative robots (or *cobots*) find their way outside the more traditional industrial context. They are used for example in medical scenarios during operations or scanning of patients. Evidently, these scenarios require sufficient safety measures. In this work, we focus on the scenario of an X-ray scanner room, equipped with several cobots (mobile scanner, adjustable tabletop and wall stand) where both patients and medical staff members can walk around freely. We propose an approach to calculate a 3D safeguard zone around people that can be used to restrict the movement of the cobots to prevent collisions. For this, we rely on four ceiling-mounted cameras. The goal of this work is to develop an accurate system with minimal latency at limited hardware costs. To calculate the 3D safeguard zone we propose to use CNN people detection or segmentation techniques to provide the silhouette input needed to calculate a 3D visual hull. We evaluate several state-of-the-art techniques in the search of the optimal trade-off between speed and accuracy. Our research shows that it is possible to achieve acceptable performance processing four cameras with a latency of 125ms with a precision of 54% at a recall of 75%, using the YOLACT++ model.

1 INTRODUCTION

In industrial processes, steady growth in robotics has led to faster and more precise manufacturing, decreasing the requirement of heavy human labour. These industrial robots often execute a preprogrammed repetitive task. However, more recently such robots are also employed outside of an industrial context, and - instead of a fixed preprogrammed task - they work together with a human operator in a cooperative manner. Hence, they are often referred to as *cobots* (Edward et al., 1999; Peshkin and Colgate, 1999; Villani et al., 2018). Even though these *cobots* are supervised and controlled by a human, important safety precautions must be taken into account to e.g. avoid collisions. In this work, we propose a vision-based safety system, which automatically calculates a safeguard zone around people in real-time. This safeguard zone can be used as an off-limits zone for the *cobots*, restricting their movements so they are unable to collide with a person present inside the robot's movement space. Our system is able to calculate this real-time person 3D safeguard zone using several multiple viewpoint cameras as input. Our method uses visual data, which nowadays is cheap, easily expandable in numbers, and capable of being processed both centralised and decentralised.

Figure 1: Use case example: Scanner room equipped with the mobile scanner, a bucky and scanning table.

To develop our system, we focus on a specific real-life clinical scenario: an X-ray scanner room with several *cobots* installed in it. In this scanner room, both patients and medical staff members are able to walk around freely. By calculating an off-limits zone automatically, we can prevent the robot from colliding with all people present, ensuring their safety.

An example of a scanner room, with all the previously described equipment installed is illustrated in figure 1.

Current safety measures only consist of a dead man's switch, operated by the medical staff. Whenever a collision is imminent, the switch is released freezing all motor functions in the room. This

Callemein, T., Van Beeck, K. and Goedemé, T.
Multi-view Real-time 3D Occupancy Map for Machine-patient Collision Avoidance.
DOI: 10.5220/0010151906270636
In *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2021) - Volume 4: VISAPP*, pages 627-636
ISBN: 978-989-758-488-6

method, however, heavily relies on the presence and awareness of the staff member. Our goal, therefore, is to automatically calculate a 3D safeguard area that can be used as an off-limits zone for the *cobots*, resulting in a much safer environment.

Such a safeguard, however, should meet stringent criteria to be usable in practice. Evidently, a high accuracy should be achieved at a low-latency performance. Due to the safety aspect, a higher recall should be prioritised over a high precision: it is better to unnecessarily stop the robot, than to stop the robot too late or not at all. Furthermore, the room lighting conditions can vary greatly, especially when the room has windows.

In a nutshell, our approach calculates an occupancy map containing the voxels of all people present in the room, to be used as a reference of positions that are inaccessible for any robotic component. For this, we rely on multiple cameras installed in the scanner room at strategical locations (e.g. four cameras placed at each ceiling corner of the room). The cameras are positioned in such a manner to have a visual overlap of the safeguarded area, allowing us to calculate 3D positions from multiple 2D detections. To generate the 2D detections, we compared a number of state-of-the-art object detectors, including both bounding box and instance segmentation.

Note that our approach is easily generalisable to other cobot applications. In this work, we employ the X-ray scanner room as a challenging, real-life application. Furthermore, the detector in our approach can easily be extended to other objects than people.

To summarize, our main contributions are:

- We developed a flexible and fast multi-view vision-based system capable of calculating a 3D safeguard zone for person-cobot collision avoidance.
- We compared both bounding box producing detectors and instance segmentation techniques as input for a visual hull calculation.
- We performed extensive experiments to determine the optimal speed and accuracy trade-off, using different state-of-the-art people detectors.

We tested the proposed approach in a real-live lab setting, and for evaluation we used a public dataset CMU (Joo et al., 2015), containing point cloud ground truth of various scenarios taken from many calibrated camera perspectives.

The remainder of this paper is structured as follows. Section 2 discusses various techniques proposed in literature to calculate a 3D representation of objects. Section 3 follows, describing our test dataset, and specifying which sequences were used during evaluation, alongside a description of the preprocessing techniques we developed. Our proposed approach is detailed in section 4, followed by section 5 discussing our results on the test datasets. We end with a conclusion and future work in section 6.

2 RELATED WORK

One of the primary concerns involving *cobots*, is the safety of the operator (Vicentini, 2020; Villani et al., 2018). When working nearby robotic parts, an emergency button must be available at all times. However, during a manufacturing process when something goes wrong, it might take some time before the operator can use the emergency button. Automatically triggering an emergency stop reduces this delay, increasing the safety of the operator. Several sensing techniques exist today, e.g. a torque sensor that measures movement resistance might trigger an error when too much force is required (Phan et al., 2018). However, these sensors only act when a collision has occurred which is not ideal and might scare the patient. Other techniques use capacitive or laser tactile proximity sensors (Navarro et al., 2013; Safeea and Neto, 2019), stopping an imminent collision between the operator and the robot only nearly before it happens. In our use case, the patients and medical staff are untrained and therefore unaware of how close the robot comes before stopping. Furthermore, only stopping when near something might still result in a crash depending on the configured proximity distance of the sensors.

Instead of mounted sensors on each mobile robotic component, Mohammed *et al.* (Mohammed et al., 2017) installed two depth cameras nearby the *cobot* and operator. By using two kinect sensors they calculate a 3D occupancy grid, enabling a safeguard zone of the people present. However, they rely on prior background data to filter out the 3D noise and known robot position to filter out the person points. In our case, this technique is not possible since people walk around in the room in addition to having no static background to filter out 3D noise. Furthermore, Mohammed *et al.* (Mohammed et al., 2017) currently only uses two depth sensors placed nearby the operator and the small robotic arm offering little chances of occlusions by other objects or people. Whereas our application, a large-sized scanner room equipped with large mobile equipment, has a higher chance of occlusions on certain cameras. To overcome this, more than two cameras can be installed capturing an overview from multiple viewpoints which might be combined to partially overcome camera occlusions. However, increasing the number of depth cameras

like the kinect comes with an increase in complexity and increase of the required computational power and hardware cost. Instead, we chose cheaper RGB sensors allowing an upscale with a feasible price, which will result in a less complex setup because the possibility of hardware sync triggering.

Most research requiring the 3D positions of people, often use a 3D skeleton-based representation. Whereas some techniques aim to calculate 3D pose keypoints (Sarafianos et al., 2016; Nie et al., 2017) from a single camera image, others use multi-view (Slembrouck et al., 2020) combining 2D pose keypoints together. The current state-of-the-art in both single view and multi-view 3D pose estimation techniques achieve real-time speed results with acceptable accuracies for their use cases (Slembrouck et al., 2020; Sarafianos et al., 2016; Nie et al., 2017). However, these techniques only output pose keypoints, whereas for our application we require a 3D bounding volume. Furthermore, both state-of-the-art techniques still have a joint position error of around 5cm, which for our application is not feasible.

Techniques like (Shi et al., 2020; Yoo et al., 2020) (evaluated using (Geiger et al., 2012)) show good performance when trying to directly estimate a car and pedestrian 3D bounding box. The best performing technique (Shi et al., 2020) uses a 3D RCNN with available LIDAR point clouds to calculate the 3D bounding boxes around objects. Although this additional sensory data is easily acquired from a vehicle perspective, in our case where we capture from a top-down perspective, occlusions might reduce the performance greatly. Furthermore, we require a more tight 3D enclosure around the person, whereas a 3D bounding box might be overestimating the person, restricting the movement of the *cobots*.

A classic method called visual hull (Laurentini, 1994) is capable of acquiring a 3D voxel grid of an object, using the silhouette of the object taken from multiple perspectives. (Abdelhak and Chaouki, 2016; Matusik et al., 2000; Vlasic et al., 2008; Furukawa and Ponce, 2006; Esteban and Schmitt, 2004). These techniques, however, often rely on a fast background subtractor whilst controlling the environment background and lighting to improve the quality of the acquired foreground, i.e. the silhouette. The mobile nature of the *cobots* might cause them to be mistaken for people by the background subtractor. To overcome the aforementioned challenges, we propose to use object detection techniques as input for such a visual hull approach, ensuring that our system works under various lighting changes and that the resulting safeguard zone only includes people.

Object detectors in most cases output a bounding box around the object, which for many use-cases is enough. Multi-stage object detectors (e.g. (Ren et al., 2015)) achieve very high accuracy by first calculating box proposals, and then performing box classification. However, the use of multiple stages increases computational complexity, rendering it difficult to achieve real-time performance. Single-stage approaches (Liu et al., 2016; Lin et al., 2017; Redmon and Farhadi, 2018) outperform the multi-stage techniques in terms of speed, with only a minor decrease in accuracy. Increasing the speed performance even further with only minor decreases of the accuracy is often achieved by changing the neural network backbone calculating the image features. For example, the recently proposed MobileNetv3+SSD (Howard et al., 2019), has a MobileNetV3 backbone optimised for embedded platforms which minimises the number of parameters and therefore the required computational cost.

The bounding boxes produced by these object detection approaches from multiple viewpoint cameras already allow to calculate a coarse visual hull. However, exact segmentation of the persons in the image evidently increases the overall accuracy of the system, since bounding boxes often tend to give an overestimation of the 3D space. Techniques like (He et al., 2017; Cai and Vasconcelos, 2019) add an additional stage after the multi-stage bounding box object detectors to generate an instance mask. However, adding an additional stage will decrease the network speed even further. A recent technique called YOLACT++ (Bolya et al., 2019b; Bolya et al., 2019a) aims at single shot instance segmentation by simultaneously detecting the bounding box and proposing mask prototypes of each object in parallel. This ensures real-time performance at the cost of only a small drop in accuracy.

In this work, we will search for the optimal trade-off between speed and accuracy by comparing both the calculated 3D safeguard zones using the bounding box detections from the MobileNetv3+SSD (Howard et al., 2019) method (in its *large* and *small* versions) against the instance segmentations from YOLACT++ (Bolya et al., 2019a). We compare the results also to more classical background subtraction techniques.

3 DATASET

To evaluate our system we require a public dataset with people in various poses in additions to occlusions, all taken from multiple calibrated top-down camera perspectives. In our use case, we mainly fo-

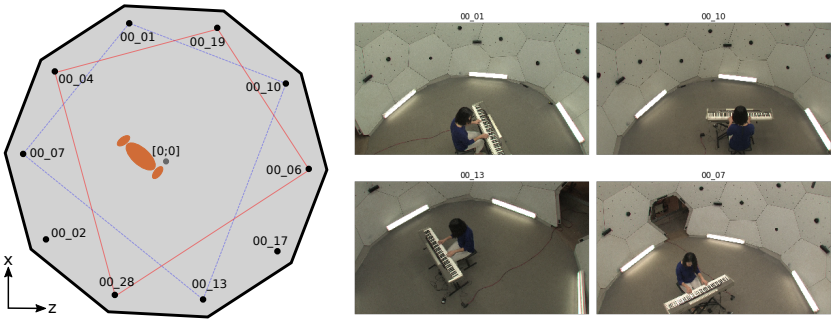


Figure 2: (left) Top-down scheme of the Panoptic dataset, showing the used cameras and two camera sets, (blue and red). (right) Example frames from the piano sequence taken from the blue camera set.

cus on the 3D position of a single patient, walking around the scanner room to take place on or in front of the table or bucky. In this room, multiple top-down wide-angle image sensors must be installed, capturing the area accessible by the patient. In addition to sensory data, person point cloud ground truth is required to measure the accuracy of our calculated patient occupancy map. We found two publicly available dataset resembling our use case best, the Panoptic Studio (Joo et al., 2015; Joo et al., 2017) and Multi-View Operation Room (MVOR) dataset (Srivastav et al., 2018). While the MVOR dataset features an operation room with similar equipment as our scanner room, too many people are present, with only a limited amount of 3D poses and movement variations. Furthermore, the dataset has only images taken from three cameras, with no person point cloud data. The Panoptic dataset, however, contains many different scenarios and pose variations of both single and multiple people, taken from different viewpoints. Although 3D point clouds acquired by the kinects are available, they are automatically generated and include noise and other objects apart from people. Below, we describe which sequences we used, followed by our pre-processing techniques to filter out only the person point clouds.

3.1 Sequences

The Panoptic dataset contains many different situations and sequences. As mentioned before, our application mainly focuses on avoiding collision with a single patient. To test various situations, we composed three subsets composed using sequences taken from the Panoptic studio dataset. Each of them will test a different scenario and will for the remainder of this paper be referred to as, the *single*, *piano*, and *multi* set. Table1 shows which sequences were aggregated from the Panoptic dataset.

The "single" set contains four sequences, each containing a single person moving around with various poses. We subsampled the large sequences in time (1 frame out of 10), since there is only little variation between frames.

Table 1: Used sequences from the Panoptic dataset.

Set	Sequences	Frames	People
Single	171026_pose1	1922	1
	171026_pose2	1412	1
	171204_pose1	2891	1
	171204_pose2	1139	1
Piano	161029_piano1	278	2
	161029_piano2	1295	1
Multi	170407_haggling_a1	2489	3

In our scanner room, the patients sometimes might be partially occluded (e.g. by the measurement instruments or a wheelchair). Such exact situations are not included in the Panoptic dataset sequences. However, some sequences show a pianist whose body is indeed partially occluded by her instrument, which we used to simulate occluded patients (the "piano" set). While other sequences with other interaction objects are available, they are not stationary and therefore difficult to exclude from the ground truth point clouds, explained in more detail in section 3.3.

The "multi" set shows multiple people walking around in the small room, frequently going outside the field-of-view of several cameras.

3.2 Camera Selection

To minimise occlusions and maximise the field-of-view in the scanner room, the best option would be to place the cameras in each corner of the scanner room, providing a top-down overview.

The Panoptic dataset is recorded in a sphere-like room with various types of cameras positioned in various locations (see figure 2). Ten wide-angle cameras are installed at the top around the room providing a top-down perspective. At the left of figure 2, a scheme of the Panoptic setup is visible, with the approximated locations and names of the cameras used in this work.

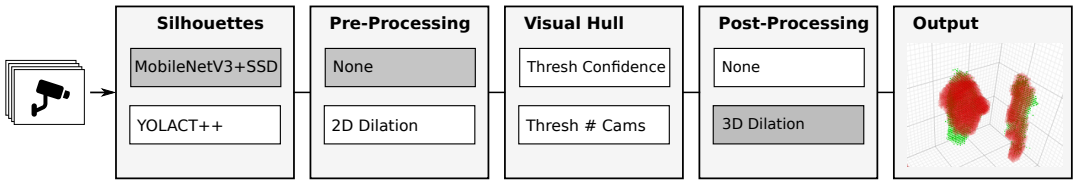


Figure 3: Our proposed approach showing the four input cameras, each used component, the pipeline output (red) and the pre-processed ground truth (green).

From these 10 cameras, we select a set of four cameras (blue) in such a way that they mimic the positions of the cameras in our scanner room, (example frames at the right-hand side of Figure 2).

During the evaluation, all nine camera combinations using these four relative camera positions are used so no camera combination is arbitrarily chosen, (e.g. next combination in red).

3.3 Pre-processing Ground Truth

A common problem when working with 3D data points is that the sheer amount of data increases the required computation power very quickly. The ground truth currently contains fine 3D positions with a high resolution, which is not required for our safeguard system. Therefore, we quantize the points to a resolution of 5cm, reducing the number of points greatly, which leads to a lower latency (due to the decrease in computational power).

As we mentioned, we require the ground truth point clouds of all people present in the room. However, since these point clouds were automatically generated using Kinect cameras solely based on captured depth maps, other objects are present in these point clouds. Therefore, in our second pre-processing step, we filter out the people points using the available annotated 3D poses.

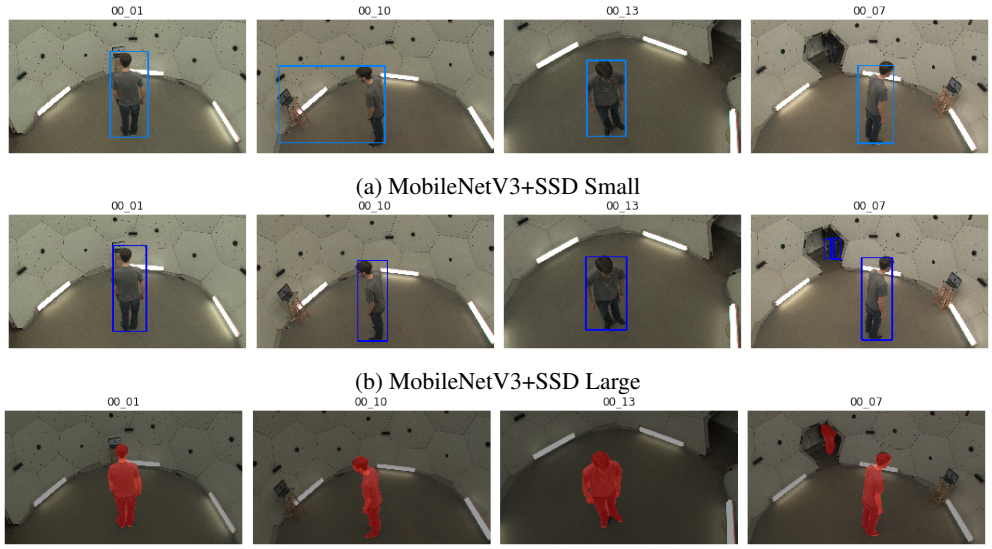
Since the point clouds were automatically generated based on Kinect depth maps only, the 3D person point clouds are hollow inside. The lack of these points poses no problem for a robot path planner since the outer points will shield the inner points. However, when comparing our generated 3D occupancy maps to the ground truth, it will seem to have a decreased accuracy due to these hollow regions. Therefore, we fill the hollow upper body region, using the ground truth 3D pose points of the neck and waist. These 3D points are dilated once in 3D, creating a 3D volume that we add to the ground truth point cloud to fill the hollow upper body region.

4 APPROACH

In the previous section, we discussed the Panoptic dataset, providing calibrated cameras images from various positions and the pre-processed 3D point cloud ground truth of each person. Our main goal is acquiring a 3D safeguard zone that makes it possible to restrict the movements of robotic parts in the scanner room, achieved by calculating a 3D people occupancy map. Figure 3 shows a block diagram of our complete approach. As input, we use four different viewpoints (i.e. cameras). In a first step, we calculate the silhouettes of each person in the image. These silhouettes are optionally pre-processed with a 2D dilation before the visual hull is calculated. Next, this visual hull can be post-processed with a 3D dilation before being used as a 3D occupancy map. In the next subsections, we describe each block in more detail.

4.1 Silhouettes

Since our specific use-case involves person safety aspects, the latency should be minimal. Furthermore, recall is more important than precision. Indeed, it is much more costly to miss a person (which might get hit by the robotic arm), then to generate a larger area where the robot cannot be used. As a latency starting point, we chose to use the MobileNetV3+SSD detector (both the small and large model) (Howard et al., 2019). These models are heavily optimised for mobile devices with low computational power and therefore have a small latency. However, this framework outputs bounding boxes. Using a bounding box instead of a silhouette to calculate the visual hull will yield over-estimating the person's 3D volume. Therefore, we compare this with the single-shot instance segmentation technique(Bolya et al., 2019a), trained to output the masks of detected objects. Figure 4 illustrates an output example of these three different models on a single time frame from four viewpoints from the Panoptic dataset. These visual results already reveal interesting observations. Visually comparing the small MobileNetv3+SSD model output(fig. 4a), with



(a) MobileNetV3+SSD Small

(b) MobileNetV3+SSD Large

(c) YOLACT++

Figure 4: Example output detections of the different frameworks.

the large model (fig. 4b), we notice that in almost all cases all persons are found. The bounding box from the small model is sometimes overestimated, and the small person in the entry of the dome was not found. Whereas the large model had a better detection rate, the bounding boxes are more accurate and the small person in the entry was found.

Because the overestimated detection only occurs on a single frame, and the person in the entry is actually not part of the ground truth. The output of YOLACT++ (fig. 4c), is capable of detecting the object bounding box along with the instance segments of each person with high confidence. However, YOLACT++ does not use pixel classification to output these instance segmentation but uses prototype masks to aggregate the single segmentation mask. Each prototype contains both positive segmentation pixel areas that are part of the object, and negative pixels areas (background or a part of another object). Together with the prototype mask, mask coefficients are calculated that combine all the prototypes to either agree or disagree together creating a full instance segmentation mask. Therefore, the contours might have a little offset from the actual person contour.

4.2 Visual Hull

We propose to construct the 3D occupancy map as follows. Firstly, Our case requires a minimum resolution of 5 cm (see section evaluation 5.1 for more details on the required specs), thus the resolution of the occupancy map is reduced to 5cm. The total safeguard 3D voxel grid contains 600.000 voxels, which are by default unoccupied. Next, we determine which of these points are occupied (by persons) by combining the silhouette output from multiple top-down perspectives, as determined above. For each camera viewpoint, we calculate the projection cone of this camera. Where it intersects with a silhouette, we increment the corresponding value for that voxel. This way, the final voxel grid values represent the number of cameras that contained a projected point of a silhouette. This value, together with the minimal required cameras, can be varied to output the 3D occupancy map (see section 5.2).

4.3 Pre-and Post-processing

As explained before, the instance segmentation contours often have a slight offset from the actual person silhouette. This implies that some 3D projected points (that should be part of the silhouette) fall outside of the contour around the person. We quantized our grid to a resolution of 5cm to decrease the number of 3D points and to increase the processing speed. However, the quantization of the points can cause some projected 3D points positioned near the contour to either be shifted inside or outside of the person silhouette. We tested two different approaches to reduce the aforementioned effects. We can either dilate the 3D occupancy map or perform a dilation of the silhouettes output from YOLACT++ (i.e. in the 2D domain). The latter is done by adding margins near the contours, which allow for more projected points to fall within the person detection silhouette. When comparing both approaches, they both showed an in-

crease in recall, however, the 2D dilation is far less computationally expensive. Furthermore, the time required to execute 2D dilation only slightly depends on the number of detections, while the execution time of the 3D dilation highly depends on the number of 3D points. Therefore, we use the 2D dilation over the 3D dilation.

5 EVALUATION

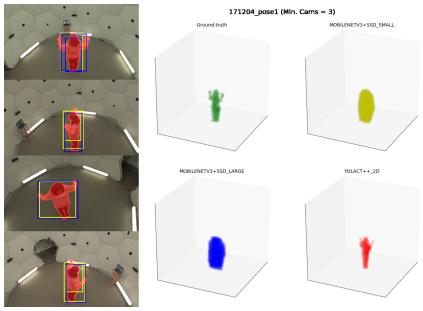
For our use case, a 3D safeguard system capable of preventing collisions with people in an automated X-ray scanner room, we search an optimal trade-off between speed and accuracy. This section first specifies the minimum requirements for such a system devised together with a manufacturer of X-ray scanner rooms, followed by the qualitative results of our approach on a single frame and video. Next, we will quantitatively evaluate the accuracy, and discuss the accuracy-speed trade-off. Finally, we will discuss the robustness of our framework against occlusions.

5.1 Specifications

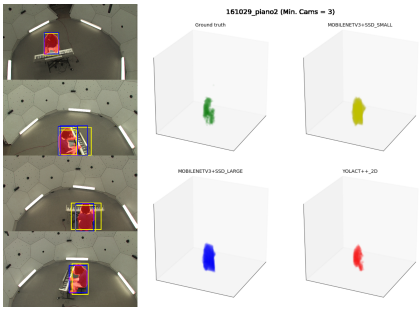
Experts in the field indicate a minimal speed of 5FPS, in other words, the 3D safeguard output of the system has a maximum allowed latency of 200ms. Furthermore, the 3D outputs must have a resolution of 5cm. Such latency and resolution allow for optimal robot control while assuring maximal safety. As explained above, we prefer high accuracy and give priority to high recall over a high precision.

5.2 3D Map

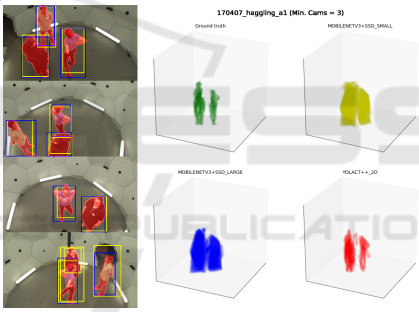
To compare the accuracy of our approach we compare the calculated safeguard voxel grid with the pre-processed ground truth. For each voxel in the ground truth, we check whether it is found in the safeguard voxel grid, producing a true positive. If this is not the case this will produce a false negative. Finally, all safeguard voxels that were not present in the ground truth are counted as false positives. We sweep over the threshold on the detection confidence of the bounding boxes and silhouettes, using the previously mentioned metrics to calculate precision-recall curves which allow us to define an optimal point, as shown in figure 6. Instead of determining an optimal point, we use the precision at a minimum recall of 0.75 as a metric to compare the different models and pre- or post-processing techniques. We used the same method to evaluate the influence of different minimum required number of viewpoints from which a person must be



(a) Single set



(b) Piano set

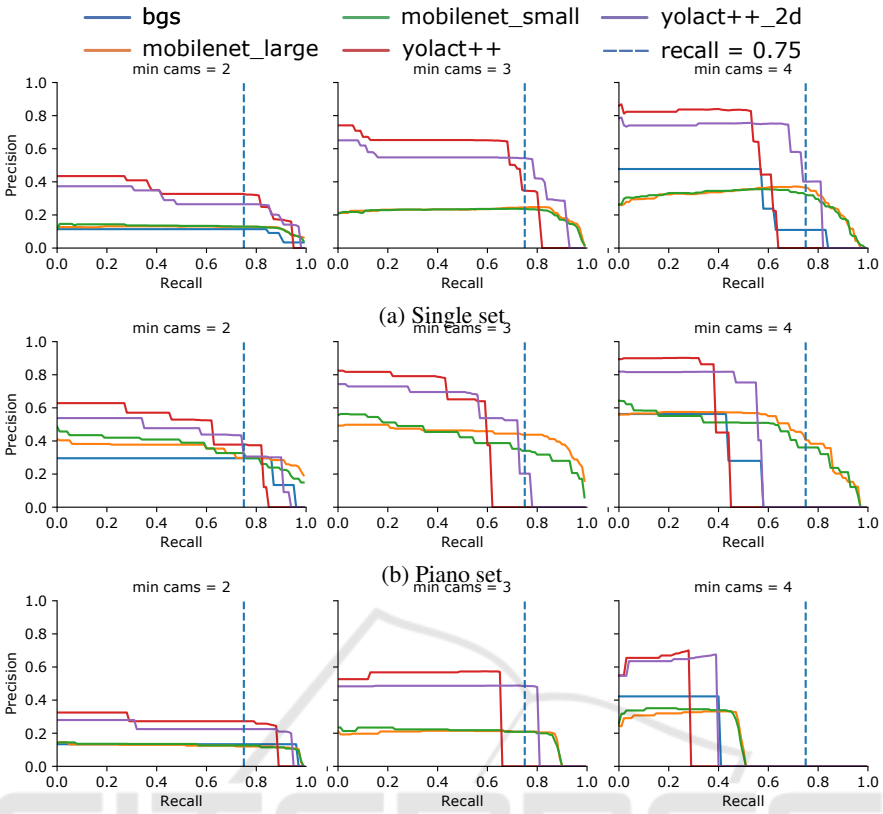


(c) Multi set

Figure 5: Example output showing the output of MobileNetV3+SSD small, large and YOLACT++ with 2D dilation.

visible. With a minimum of 2 cameras producing a higher recall with lower precision due to filtering out fewer voxels. Whereas a min. of 4 cameras is more strict with a lower recall and higher precision. From the 10 different camera viewpoints, we consecutively select a set of four relative camera positions (see fig. 2 for an example of two sets - red and blue). In total. we thus evaluate 10 different sets of camera positions for each frame. A single-precision result is calculated by using the micro-average of all 10 sets. Figure 5 shows a qualitative evaluation for each test set (with minimum 3 cameras), showing the detections on the camera frames along with the 3D ground-truth and output for each model[1]. Both figure 5a and 5b show the output of a single person, clearly indicating that

[1]Full video: https://youtu.be/n-HfHBgd-EI



(a) Single set

(b) Piano set

(c) Multi set

Figure 6: PR-curves of each model for the different subsets.

the 3D output of YOLACT++ based silhouettes (red) is finer, compared to the bounding boxes approach (yellow and blue). However, bounding boxes yield a more coarse 3D estimation, which is to be expected an overestimation of the volume of the person. For the application at hand, this means that these methods will produce a wider safeguard zone around the persons, hence a better recall (but worse precision) as will be demonstrated below. Figure 5c shows a similar behaviour, with less space between people near each other on the bounding box method compared to the instance segmentation approach.

5.3 Precision vs. Speed

Figure 7 display the measured performance of the single, piano and multi test sets, showing the latency versus the precision (with a set minimum recall of 0.75) for each model. Each configuration is represented by a circle, with the colour representing the used detection method. The size of the circle represents the set required minimum number of viewpoints that contributed to the voxels. All these experiments measuring latency were executed on an i7-8750H with 32 GB RAM with an RTX 2060 GPU.

As a baseline method to compare against, we also used silhouettes procured by a Mixture of Gaussians background subtraction background subtractor (BGS) approach with an image resolution of 480×270 (Zivkovic, 2004; Zivkovic and Van Der Heijden, 2006). Although we expected the MobileNetV3+SSD models (using an image resolution of 224×224) to be the best detection based approach, in terms of latency, the large model seems to be almost 5% slower than the YOLACT++ model with a larger input resolution of 550×550. In terms of performance, we show results as comparison in figure 6, showing that YOLACT++ far outperforms both the BGS and MobileNetV3+SSD. For MobileNetv3+SSD this is mainly caused by the overestimating of the bounding box silhouettes, causing many false positives. With the BGS approach, we see that certain body parts are missing, which required us to add sufficient dilation to reach the minimum recall of 75%. Moreover, using background subtraction it is unavoidable that people disappear in the background when immobile, which is in our application always the case as patients are laying on a table or standing still during scans. The BGS results on this Panoptic dataset hence show a better performance than what is expected in a real scanning

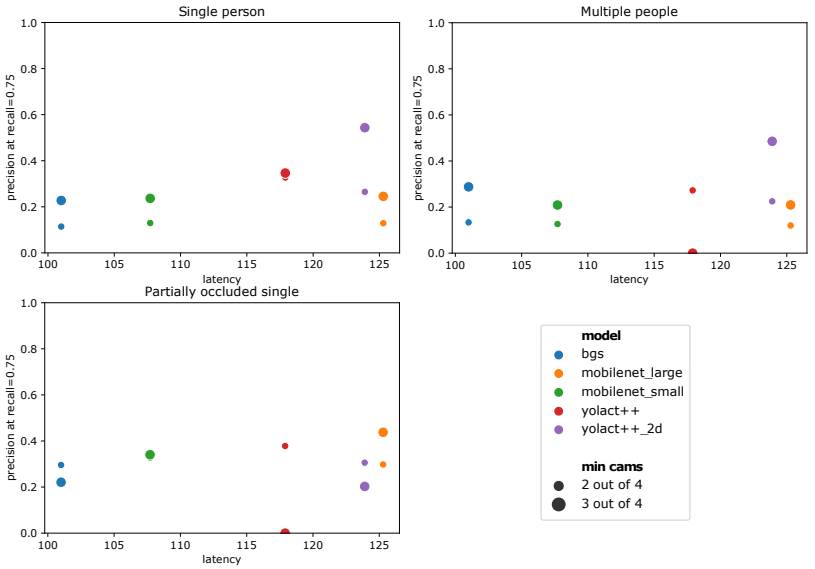


Figure 7: Latency vs. precision at a minimum recall of 0.75 for all sequences.

room. In the case of the Yolact++ approach, with no missing body parts, adding the 2D dilation causes an increase in both recall and precision.

5.4 Occlusions

In a second experiment, we evaluate the performance of our framework with regard to occlusion. As seen for the single test set results (top-left graph of fig. 7), our framework achieves good performance in both speed and accuracy. However, for sets with occlusion (*piano set* and *multi set*, a drop in precision and recall is seen. Figures 6c and 6b show that a decrease occurs when the minimum number of required viewpoints is set to 4 (i.e. all 4 cameras need to find the detections), even causing some approaches to not reach the minimum recall of 75%. This threshold is considered very strict since any voxel not projected within a detection on all four cameras is filtered out. Depending on the level of occlusion, this is to be expected since missing parts will not be compensated for by the other cameras with no occlusion. Hence, our approach enables us to create a safeguard zone, even around partially occluded people, by setting this amount of required views lower than the number of cameras installed.

6 CONCLUSIONS

In this work, we searched for a detection based approach capable of calculating a 3D safeguard region to ensure person safety by restricting the movements of *cobots* in e.g. medical scanning rooms. In this paper, we proposed to extend the classic visual hull 3D estimation technique with CNN-based person detection and segmentation methods, instead of the traditionally used background subtraction. We evaluated several techniques on a public dataset comparing their latency and precision at a guaranteed recall. Our results show that the 2D dilated Yolact++ approach reaches a precision of 54% with a recall of 75% with a latency of 123ms. Even though the latency is higher compared to a traditional BGS, it achieves higher precisions and still performs faster than the maximum latency of 200 ms. In future work, a dataset featuring the actual equipment could be gathered to evaluate even further, adding the challenge of mobile *cobots* causing more moving occlusions that are disastrous for BGS.

ACKNOWLEDGEMENTS

This work is supported by VLAIO and AGFA NV via the Start to Deep Learn TETRA project.

REFERENCES

Abdelhak, S. and Chaouki, B. M. (2016). High performance volumetric modelling from silhouette: Gpu-image-based visual hull. In *IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*. IEEE.

Bolya, D., Zhou, C., Xiao, F., and Lee, Y. J. (2019a). Yolact++: Better real-time instance segmentation.

Bolya, D., Zhou, C., Xiao, F., and Lee, Y. J. (2019b). Yolact: Real-time instance segmentation. In *ICCV*.

Cai, Z. and Vasconcelos, N. (2019). Cascade r-cnn: High quality object detection and instance segmentation. *arXiv preprint arXiv:1906.09756*.

Edward, J., Wannasuphoprasit, W., and Peshkin, M. (1999). Cobots: Robots for collaboration with human operators.

Esteban, C. H. and Schmitt, F. (2004). Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*.

Furukawa, Y. and Ponce, J. (2006). Carved visual hulls for image-based modeling. In *European Conference on Computer Vision*. Springer.

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*. IEEE.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*.

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al. (2019). Searching for mobilenetv3. In *Proceedings of the IEEE International Conference on Computer Vision*.

Joo, H., Liu, H., Tan, L., Gui, L., Nabbe, B., Matthews, I., Kanade, T., Nobuhara, S., and Sheikh, Y. (2015). Panoptic studio: A massively multiview system for social motion capture. In *The IEEE International Conference on Computer Vision (ICCV)*.

Joo, H., Simon, T., Li, X., Liu, H., Tan, L., Gui, L., Banerjee, S., Godisart, T. S., Nabbe, B., Matthews, I., Kanade, T., Nobuhara, S., and Sheikh, Y. (2017). Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Laurentini, A. (1994). The visual hull concept for silhouette-based image understanding. *IEEE Transactions on pattern analysis and machine intelligence*.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *ECCV*. Springer.

Matusik, W., Buehler, C., Raskar, R., Gortler, S. J., and McMillan, L. (2000). Image-based visual hulls. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*.

Mohammed, A., Schmidt, B., and Wang, L. (2017). Active collision avoidance for human–robot collaboration driven by vision sensors. *International Journal of Computer Integrated Manufacturing*.

Navarro, S. E., Marufo, M., Ding, Y., Puls, S., Göger, D., Hein, B., and Wörn, H. (2013). Methods for safe human-robot-interaction using capacitive tactile proximity sensors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE.

Nie, B. X., Wei, P., and Zhu, S.-C. (2017). Monocular 3d human pose estimation by predicting depth on joints. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE.

Peshkin, M. and Colgate, J. E. (1999). Cobots. *Industrial Robot: An International Journal*.

Phan, T.-P., Chao, P. C.-P., Cai, J.-J., Wang, Y.-J., Wang, S.-C., and Wong, K. (2018). A novel 6-dof force/torque sensor for cobots and its calibration method. In *IEEE International Conference on Applied System Invention (ICASI)*. IEEE.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*.

Safeea, M. and Neto, P. (2019). Minimum distance calculation using laser scanner and imus for safe human-robot interaction. *Robotics and Computer-Integrated Manufacturing*.

Sarafianos, N., Boteanu, B., Ionescu, B., and Kakadiaris, I. A. (2016). 3d human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding*, 152.

Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., and Li, H. (2020). Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *CVPR*.

Slembrouck, M., Luong, H., Gerlo, J., Schütte, K., Van Cauwelaert, D., De Clercq, D., Vanwanseele, B., Veelaert, P., and Philips, W. (2020). Multiview 3d markerless human pose estimation from openpose skeletons. In *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer.

Srivastav, V., Issenhuth, T., Kadkhodamohammadi, A., de Mathelin, M., Gangi, A., and Padoy, N. (2018). Mvor: A multi-view rgb-d operating room dataset for 2d and 3d human pose estimation. *arXiv preprint arXiv:1808.08180*.

Vicentini, F. (2020). Collaborative robotics: a survey. *Journal of Mechanical Design*.

Villani, V., Pini, F., Leali, F., and Secchi, C. (2018). Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics*.

Vlasic, D., Baran, I., Matusik, W., and Popović, J. (2008). Articulated mesh animation from multi-view silhouettes. In *ACM SIGGRAPH 2008 papers*.

Yoo, J. H., Kim, Y., Kim, J. S., and Choi, J. W. (2020). 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. *arXiv preprint arXiv:2004.12636*.

Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* IEEE.

Zivkovic, Z. and Van Der Heijden, F. (2006). Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*.