

# Part Alignment Identification and Adaptive Pick-and-place Operation for Flat Surfaces

Paulo Moreira da Costa<sup>1</sup>, Paulo Costa<sup>1,2</sup>, Pedro Costa<sup>1,2</sup>, José Lima<sup>1,3</sup> Germano Veiga<sup>1</sup>

<sup>1</sup>INESC TEC (formerly INESC Porto)

<sup>2</sup>Faculty of Engineering, University of Porto

<sup>3</sup> School of Engineering, Polytechnic Institute of Bragança  
paulojorgemcosta@gmail.com, {paulo.j.costa, pedro.g.costa,  
jose.lima, germano.veiga, }@inescporto.pt

**Abstract:** Industrial laser cutting machines use a type of support base that sometimes causes the cut metal parts to tilt or fall, which hinders the robot from picking the parts after cutting. The objective of this work is to calculate the 3D orientation of these metal parts with relation to the main metal sheet to successfully perform the subsequent robotic pick-and-place operation. For the perception part the system relies on the low cost 3D sensing Microsoft Kinect, which is responsible for mapping the environment. The previously known part positions are mapped in the new environment and then a plane fitting algorithm is applied to obtain its 3D orientation. The implemented algorithm is able to detect if the piece has fallen or not. If not, the algorithm calculates the orientation of each piece separately. This information is later used for the robot manipulator to perform the pick-and-place operation with the correct tool orientation. This makes it possible to automate a manufacturing process that is entirely human dependent nowadays.

**Keywords:** kinect, 3d vision, Pick-and-place, robotic manipulator

## 1. Introduction

Laser cutting machines are widely used on metallurgical industry. Even though there are different manufactures, the machines share the same basic kinematic structure, which consists of a Cartesian robot coupled with a laser that covers the entire workspace. The main metal sheet lies on a metal support base that maximizes the presence of air below the metal sheet to be cut. This is ensured using a support composed of vertical triangles where the metal lies only on its tips, Figure 1a).



**a)** Laser cutting machine [1]

**b)** Tilted piece after cut

**Fig. 1:** Laser cutting machine and cut metal piece example

With this architecture the metal parts tend to tilt or fall after the cut, making the robot's collection task more complex, Figure 1b). In order to perform the pick-and-place operation, the robot needs to perceive the misalignment of the cut parts to enable the automation of the subsequent picking operation. The use of an industrial robot for this operation requires object pose identification because the piece extraction trajectory depends on its orientation at that time.

This paper is divided in five chapters including the introduction. In the next chapter the state of the art is presented. The process to fulfil the objectives is described in chapter three. In chapter four the results are presented and discussed. Finally, in chapter five the paper is concluded and future work is proposed.

### 1.1. Objectives

The objective of this work consists of detecting the 3D position and orientation of cut metal parts in order to successfully perform the pick-and-place operation. This means that the robot has to decide if the pieces can be picked or not. In the affirmative case, the robot has to approach the metal part with correct tool orientation angle. In other words, the system has to adaptively perform the pick-and-place operation with regard to the piece to be collected, always avoiding picking the fallen ones.

## 2. State of the Art

Similar work can be found on bin picking related projects that include research on perception, grasping and path planning algorithms. Perception is the most relevant aspect in this work and, therefore, the state of the art presented here will focus only on that area. This is because the metal parts have a flat contact area that facilitates

grasping techniques. The path planning is also simplified since there are no obstacles during the pick-and-place operation. [2, 3]

The system relies on three-dimension vision hardware. These technologies can be active or passive depending on whether there is interaction with the environment or not, respectively. Due to their mode of operation and sensor characteristics these technologies can be divided as (a) triangulation based active ranging (PrimeSense technologies), (b) vision based passive ranging (stereoscopic vision), and (c) time-of-flight active ranging (laser rangefinders). Triangulation based active ranging technologies use geometric properties manifested in their measuring strategy to



**Fig. 2:** Microsoft Kinect, developed by PrimeSense

establish distance readings to objects. Vision based passive ranging technologies are sensing devices that capture the same raw information light that the human vision system uses. Finally, time-of-flight active ranging technologies makes use of the propagation speed of sound or an electromagnetic wave. [4, 5]

PrimeSense [6] is responsible for developing the Microsoft Kinect [7], shown in Figure 2, and the Asus Xtion [8]. They share the same work principles and they are known for their good performance at a considerably low price. In addition, the work requirement ranges fit the technical limitations of the Microsoft Kinect. Internally, the Kinect contains an RGB camera, an infrared (IR) camera and an IR projector. Its three-dimensional vision characteristics come from triangulation between two consecutive IR frames. It is possible to build a colorized point cloud by mapping the depth map with the information from the RGB camera. There is a wide range of research groups developing computer vision solutions based on this technology. All the facts considered make this hardware suitable for the perception subsystem implementation [9, 10, 11].

### 3. Methodology

The implemented system depends on perception and robot control. The perception component relies exclusively on the Microsoft Kinect. The image data is later processed together with the known pieces position and format information used as input data for the laser cutting machine side. This, combined with a plane fitting algorithm, returns the pick position and orientation that serves as input for the robot trajectory control.

### 3.1. Architecture

Due to the industrial nature of the project, it was necessary to simulate the working environment in the laboratory. The hardware architecture implemented is mainly divided as follows: Microsoft Kinect, ABB Robot, computer, cut metal sheet provided by a company in this field, and a wooden prototype support base, Figure 3.



a) Laboratory architecture

b) Main metal sheet

**Fig. 3:** Set of hardware used for the system implementation

A high level application is responsible for controlling the perception hardware and the data is shared with the robot via serial communication. The software solves the computer vision algorithms and uploads to the robot the calculated position and pose of the robot target.

### 3.2. Perception

The perception is responsible for solving the sensing raw data coming from specific algorithms that provide meaning to the acquired environment information. The Kinect observes the environment and builds a depth map with pixel values proportional to the object distance. To turn the raw calculated distances into SI units, the conversion method presented in Equation 1 is used, as proposed in [12].

$$d_m = 0.1236 \tan\left(\frac{d_k}{2842.5} + 1.1863\right) \quad (1)$$

where  $d_k$  is the raw depth to a specific point directly provided by the Kinect;  $d_m$  represents its conversion to meters.

Considering this dimension as  $z$  axis on the Kinect reference frame, then  $x$  and  $y$  are defined according to their width and height. These last two dimensions need to be interpolated from the depth distances taken from the Kinect, as show in Equation 2.

$$P_c(u, v) = \begin{bmatrix} x_c(u, v) \\ y_c(u, v) \\ z_c(u, v) \end{bmatrix} = d_m(u, v) \times \begin{bmatrix} (u - c_x^{IR}) / f_x^{IR} \\ (v - c_y^{IR}) / f_y^{IR} \\ 1 \end{bmatrix} \quad (2)$$

where,  $P_c$  are the coordinates in meters related to the IR camera frame;  $u$  and  $v$  are the coordinates in pixels also related to the IR camera frame;  $f$  and  $c$  are the intrinsic parameters of the IR camera ( $f$  focal length and  $c$  the distance between the lens and the focal point).

At this point, the depth map returned from the computer vision hardware contains all the pixels mapped to the camera reference frame in meters. However, it is useful to have the depth map referenced to a frame that is shared with the robot so that a specific 3D point has the same definition both for the Kinect and robot frames, that is, the world reference frame. This results in a homogeneous transformation from the camera to world reference frames, Equation 3.

$$P_w = (P_c - T) \times R \quad (3)$$

where,  $P_c$  are the coordinates in camera reference frame, and  $R$  and  $T$  are the rotation and translation matrices, respectively.

In this project, the Kabsch algorithm [13] was used in order to calculate the matrices responsible for the above mentioned transformation. This algorithm uses two sets of paired points, where one is referenced to the Kinect and the other is referenced to the world frame. Firstly, the translation is calculated by taking the centroids of the two meshes and the consequent distance between them. Both sets of points are then centred on their respective centroids. Secondly, it uses the covariance matrix to calculate the optimal rotation matrix that minimizes the root mean squared deviation error between both sets. From this point, the depth map is referenced to the same robot work frame. As a result, it is possible for the robot to work directly with the coordinates returned from the Kinect.

The Kinect acquires more than the region of interest. Therefore, after the overall environment mapping, it is advantageous to work only in the region of interest to achieve faster processing times. After transforming the coordinates, this is easily done by disregarding the points outside a specific range of values for the three dimensions in  $x$ ,  $y$  and  $z$  in the world frame.

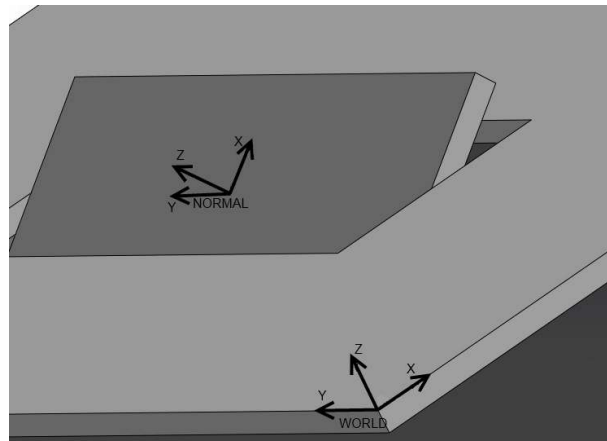
This computer vision hardware returns null reading values for points where it was not possible to calculate the distance. Since the current frame depends on the previous frame, a reading error on the previous frame affects the accuracy of the current frame for that specific point. Consequently, a simple pre-processing technique is applied in order to increase the quality of the depth map. This technique consists of calculating the median of three consecutive frames where the median depth is only calculated for 3D points where no reading errors occurred. This increases the number of null values; however, it increases the reliability of the depth map. An example of this method is presented in Figure 4.

n-2				n-1				n				result			
13	13	13	14	14	12	13	15	12		12	15	13		13	15
14	13		15	14	13		15	13	12	13	15	14	13		15
12	14	13	15	12	14	13	14	12	13	14	15	12	14	13	15

**Fig. 4:** Illustration of the pre-processing technique

### 3.3. Pose Identification

In order to calculate the orientation angle of a specific metal part, the algorithm implemented starts by matching the known piece positions to the depth map returned from the computer vision hardware. Therefore, there is no implicit piece detection based on image processing algorithms. Alternatively, since both the Kinect and the robot have the same reference frame, the known positions of the metal parts can be mapped directly onto the depth map. For the plane fitting algorithm, only a set of points is considered that match a circle whose radius is proportional to the size of the metal part. Thus, the algorithm uses a limited set of values based on one point from the previously known data, originated from the laser cutting machine design software. The mentioned group of points maps a delimited circular region for each piece that works as input for the plane fitting algorithm. Its implementation uses singular value decomposition (SVD) and returns the normal vector to the plane defined from the input points referenced to its orthonormal reference frame. With this normal it is possible to calculate the orientation magnitude between this vector and the normal of the main metal sheet. If the world reference frame has xOy coinciding with the metal sheet plane, then its normal will have the direction of z. The magnitude orientation is solved as an ordinary angle calculation between the vectors  $z_w$  and  $z_n$ , where w and n represent the world and piece (based on the normal vector) reference frames, respectively, Figure 5. The returned normal vector contains more relevant information since its projection in the world's xOy plane reveals the 2D orientation of the tilted piece.



**Fig.5:** Illustration of the world and normal reference frames

To perform a trajectory the robot needs the position and consequent orientation of the tool. More specifically, the position is a value in each x, y and z axes referenced to

some frame and the orientation is set with a quaternion. Therefore, it is necessary to have an orthonormal reference frame in each metal sheet where both the position and orientation are mapped. This is done by applying the plane fitting algorithm knowing that the world reference frame is already set in XYZw. Then it is possible to calculate the other two axes that together with the zn build an orthonormal reference frame XYZn, Equation 4 and 5.

$$y_n = z_n \times x_w \quad (4)$$

$$x_n = y_n \times z_n \quad (5)$$

With these simple cross product calculations, and using the normal vector returned from the plane fitting algorithm, it is possible to map both the position and orientation angle using an orthonormal reference frame. The frame origin maps the position in the world reference frame, and the orientation is provided by the deviation between both reference frame axes, Figure 5.

### 3.4. Robot Control

At this stage the perception algorithm provides all the input data necessary for the robot to perform the pick-and-place trajectory, that is, the calibrated world reference frame, and the position and orientation of the metal part. The pick positions with correct orientation are the robot targets in a specific robot trajectory.

The system contains all the input data necessary for the robot to perform the pick-and-place trajectory. The Kinect and the robot share the same work reference frame and the perception system is able to calculate both the position and orientation for each metal piece.

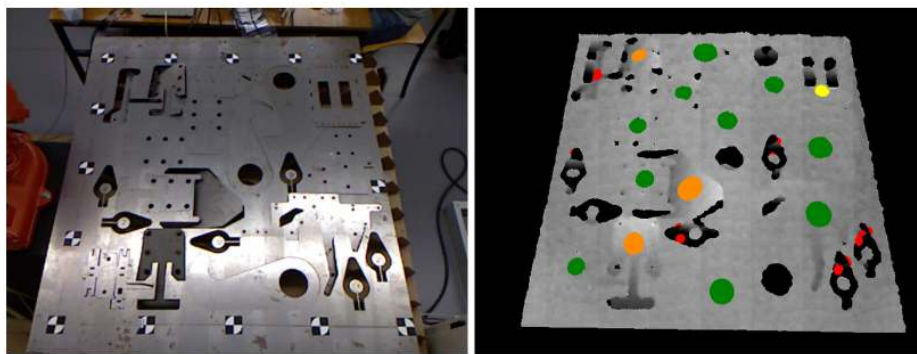
Therefore, the robot is controlled using the previously known position and calculated orientation as input data. This data is transferred from the industrial computer to the robot over serial communication. The robot receives the data and computes it iteratively for each metal part.

## 4. Results

The results will cover the two main parts of this project: perception and robot picking performance, using the previously presented architecture. Firstly, examples of the perception algorithm are presented and the results are discussed. Secondly, a number of consecutive picking operations are performed in order to numerically approximate the robot picking reliability. The tests consist of putting the cut pieces aligned with the main metal sheet and letting them rearrange arbitrarily. This simulates a normal scenario where the main cut metal parts come from the laser with unknown orientations.

Figure 6 shows two images acquired with the Kinect, both representing the same scenario directly seen from its point of view: colorized scene from the RGB camera,

Figure 6a), and depth map in grey scale with circular regions of interest for the plane fitting algorithm, Figure 6b). In the depth map, darker colours mean farther distances to the Kinect and the black colour maps points outside of the area of work or with unknown distances. In the same picture is evident the surrounded main metal sheet area which represents the area of work. This area is automatically obtained after the world reference frame calibration. The perception software classifies the pieces according to their orientation: green means alignment with the metal sheet (no orientation), orange (tilted piece) and red means invalid orientation (absent or fallen piece).



a) RGB frame

b) Depth in grey scale with circular regions of interest

**Fig.6:** View from the Kinect of an example scenario that includes tilted and fallen pieces

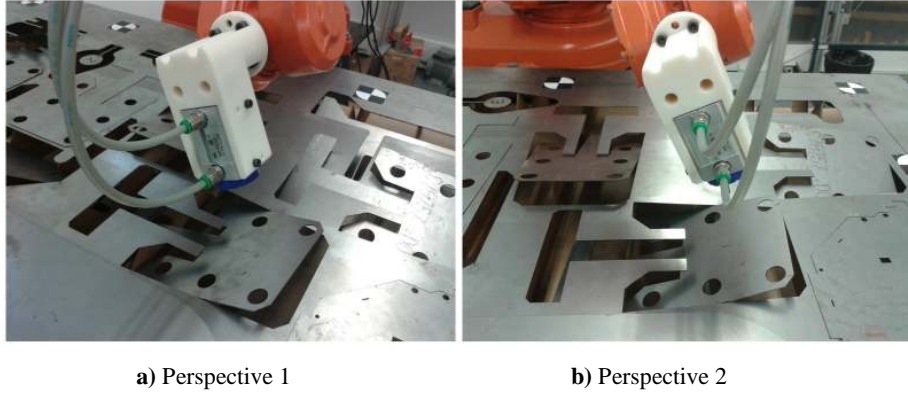
For the misaligned pieces, it was possible to compare the plane fitting results to the measurements taken from the piece itself regarding its orientation relatively to the main metal sheet. The plane fitting calculations were conducted three consecutive times to make it possible to study the repeatability performance of the algorithm. The averages of these calculations can then be compared to the measurements. This test was performed on four different pieces at increasing distances from the Kinect. This means that piece 1 is the closest (1,0m) and piece 4 is the farthest (1,70m). The results are presented in Table 1. In the last column, the standard deviation of the three plane fitting calculations shows that the repeatability decreases as the distance from the objects increases. The plane fitting calculation error is also consistent with the Kinect error dynamic because it increases proportionally to the distance. These numbers show that the distance affects both the repeatability and the accuracy as the algorithm depends directly on the performance of the Kinect. Piece 4 represents the farthest piece on the work area and, therefore, it approximates the highest error for the plane fitting algorithm. This accuracy is sufficient for the system validation, because collecting the piece with magnetic or vacuum tool has some orientation compliance. Therefore, this small error (only evident for longer distances) does not jeopardize the picking operation.



**Table 1:** Comparison between measurements and plane fitting calculations

Pieces	Measurements	Plane fitting calculations			$\mu$	$\mu$ -M	$\sigma$
1	20°	19°	19°	20°	19.3°	0.7°	0.6
2	25°	25°	26°	27°	26.0°	1.0°	1.0
3	22°	25°	27°	26°	26.0°	4.0°	1.0
4	21°	26°	27°	23°	25.3°	4.3°	2.1

The robot was coupled with a magnetic gripper, as demonstrated in Figure 7, in order to test the overall system in the laboratory test-bed. The geometry of the metal parts hinders the pick of the tilted pieces since they get stuck in the process. It is impossible for the robot to know this information a priori. The pieces are explicitly classified as aligned if they present an absolute orientation angle below five degrees, represented in green in Figure 6b). For these cases, the robot was able to successfully pick all pieces for three consecutive times in its work range without any failure. For the cases where picking is impossible, the robot was also able to successfully align with all of them, as demonstrated in Figure 7.



**Fig. 7:** Tool approach with magnetic gripper for tilted piece

## 5. Conclusion

The chosen computer vision hardware presented good performance results when calculating the depth map. The implemented conversion to SI units, associated with reference frame calibration, made it possible to easily share the results from perception hardware with the robot. The plane fitting algorithm returned accurate results for the normal vector which is accurate enough for the problem considered. This implementation shows that low cost vision hardware such as the Kinect can be used for industrial applications. The precision is sufficient even when working on its

technical limitations. The results are excellent when working for closer distances. As a consequence, the position of the Kinect should be previously studied to take advantage of its best performance.

Finally, the robot is able to perform the pick-and-place operation using the information from the perception subsystem. With the result from the plane fitting algorithm the robot can decide whether to pick, to approach or to avoid a specific metal piece. The picking of the aligned pieces demonstrated an excellent performance and the approach is also very accurate with piece orientation.

### **5.1. Future Work**

The system implemented uses the Kinect, which has a limited area of work. It would be interesting to upgrade the system to work with multiple Kinect systems or similar sensors. This would increase the area of work and also the quality of the depth when overlapping the information obtained with multiple sensors. If the sensors are positioned correctly, it is possible to avoid occlusions, thus significantly reducing null data.

### **ACKNOWLEDGEMENTS**

The work presented in this paper, being part of the Project PRODUTECH PTI (nº 13851) – New Processes and Innovative Technologies for the Production Technologies Industry, has been partly funded by the Incentive System for Technology Research and Development in Companies (SI I&DT), under the Competitive Factors Thematic Operational Programme, of the Portuguese National Strategic Reference Framework, and EU's European Regional Development Fund".

The authors also thanks the FCT (Fundação para a Ciência e Tecnologia) for supporting this work trough the project PTDC/EME-CRO/114595/2009 - High-Level programming for industrial robotic cells: capturing human body motion.

### **REFERENCES**

1. Adira, <http://www.adira.pt>
2. Song, K.-Tai, Tsai, S. : Vision-based adaptive grasping of a humanoid robot arm, Automation and Logistics (ICAL), 2012 IEEE International Conference on , vol., no., pp.155-160, 15-17 Aug. 2012
3. Pinto, M., Moreira, A. Paulo, Costa, P. , Ferreira, M., Malheiros, P. : Robotic manipulator and artificial vision system for picking cork pieces in a conveyor belt. 10th Conference on Mobile Robots and Competitions, Robotica 2010.
4. Siegwart, R. , Nourbakhsh, I. : Introduction to Autonomous Mobile Robots. Bradford Company, Scituate, MA, USA, 2004.
5. Moreira da Costa, P. : Operação de “Pick-and-place” Adaptativo para Ambientes Pouco Estruturados. Master Thesis, 2012.
6. PrimeSense official site. <http://www.primesense.com>
7. Microsoft. Xbox 360 kinect. <http://www.xbox.com/kinect/>
8. Asus Xtion, [http://www.asus.com/Multimedia/Motion\\_Sensor/Xtion\\_PRO/](http://www.asus.com/Multimedia/Motion_Sensor/Xtion_PRO/)

9. Technical description of Kinect calibration.,  
[http://www.ros.org/wiki/kinect\\_calibration/technical](http://www.ros.org/wiki/kinect_calibration/technical)
10. Khoshelham, K. , Elberink, S. : Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications, In Proceedings of Sensors 2012, pp.1437-1454, 2012.
11. Khoshelham, K. : Accuracy Analysis of Kinect Depth Data, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XXXVIII-5/W12, 133-138, doi:10.5194/isprsarchives-XXXVIII-5-W12-133-2011, 2011
12. Image information. [http://openkinect.org/wiki/Imaging\\_Information](http://openkinect.org/wiki/Imaging_Information)
13. Kabsch, W. : Automatic processing of rotation diffraction data from crystals of initially unknown symmetry and cell constants. J. Appl. Cryst. 26, 795-800, 1993.