Vision Based Detection of Butt Weld Joints for Robotic Arc Welding

Abhishek P G
Department of Mechanical Engineering
Government Engineering College
Thrissure, India
abhishekpg01@gmail.com

Dr. Lalu P P
Nodal Centre for Robotics and AI
Government Engineering College
Thrissure, India
lalujesus@yahoo.co.in

Dr. Sudheesh R S
Nodal Centre for Robotics and AI
Government Engineering College
Thrissure, India
rssudheesh@gectcr.ac.in

***Abstract*— One of the main problems for robotic welding is the time taken to programme the weld path for a new job. Currently, robotic welding is programmed through 'teach and playback method'. In this method, the waypoints (3D co-ordinates) of the weld seam is manually fed into the robot controller. For each new part, the waypoints need to be updated. Also, the waypoints need to be changed if the relative position of the weld joint is changed. This will increase the set-up time of the robotic welding. In order to solve this problem, computer vision techniques can be used to find out the waypoints automatically, instead of feeding it manually to the robot controller. However, the welding environment presents certain challenges for computer vision. These challenges include poor contrast, reflections from metallic surfaces, and imperfections on the work piece such as rust, mill scale and scratches which are not consistent from part to part. This paper presents a new method that can detect the butt weld joints autonomously using vision sensors. The flexibility of the proposed method is tested under three test cases which are; (1) Straight vertical, (2) Straight horizontal, (3) saw tooth and (4) right angle butt joints. The results show that the proposed method is capable of detecting the butt weld joints irrespective of shape and orientation of the weld joint.**

***Keywords*— Welding, Robotic Welding, Computer Vision**

I. INTRODUCTION

Welding is an important and fundamental fabrication method in manufacturing. The lack of skilled welding labour, need to increase production and increase safety in the workplace have increased demand for robotic welding. Most existing robotic welding systems uses 'teach and playback' method. In 'teach and playback' method a manual teaching pass is require before actual welding. This method is time consuming and the welding robot needs to be reprogrammed each time the workpiece of the weld is changed. In order to solve this problem, sensors need to be integrated with robot system to detect the weld path autonomously.

There are several types of sensors used for weld detection. [1] discusses commercially available weld tracking sensors and its applications in robotic welding. Researchers in [2] used a laser displacement sensor to detect the weld seam in double hulled structures in shipyards. The weld is detected by scanning the laser sensor over the seam profile to be welded. Authors of [3] discusses the merits and demerits of different types of sensors used for robotic welding. Currently available sensors for detecting the weld seam include Laser sensor, ultrasonic sensor, vision sensor, electro-magnetic sensor etc. The most widely used sensor is vision sensor due to its advantages over other available sensors. Vision sensor has more advantages compared to other sensors such as fast speed, high precision, non-contact and low cost.

In most of vision-based weld detection algorithms for butt welding, the first step is to eliminate the background from the captured image to ensure that the weld detection algorithm is not affected by welding environment. Researchers in [4] used different image processing techniques such as smoothing, sharpening and region segmentation for segmenting the work piece from the background. [5] describes a background elimination algorithm to eliminate the effect of complex robotic welding environments. An image of the background without the work piece is pre-loaded to eliminate the background. The captured image is compared with the pre-loaded image of background to eliminate the background. Researchers in [6] and [7] uses boundary detection using Hough transform to detect the outside boundary of the weldment.

Another method to eliminate the background from the captured image is to use Region of Interest (ROI) which is a sample portion of the captured image. Authors in [8] uses a rectangular ROI at the centre of the image to separate the background. The image is captured in such a way that, the workpiece is located inside the ROI rectangles and the environment is located outside of the ROI. In [9] uses a local ROI by assuming that, a small portion of the weld joint is included in the ROI. The remaining weld joint outside the ROI is detected by extending the weld joint inside the ROI by edge linking method.

Once the background is eliminated, the next step is to detect the weld joint within the weldment. [10] describes an improved Ghosal algorithm to detected the weld edges more accurately based on Zernike moments. Meanwhile in [11], a visual servoing method is used to detect the seam coordinates for robot arc welding. Researchers in [12] has developed an online weld tracking system using improved canny edge detection algorithm. A detailed review on vision-based weld tracking methods is given in [13]. The advantages, drawback and limitations of different approaches are discussed.

In this paper, a new weld joint detection method for butt welding is introduced. The proposed algorithm is capable of detecting the butt weld joints of any shape and orientation. The methodology used in weld detection is discussed in detail in the following session..

II. METHODOLOGY

In this paper, a weld detection method is introduced. It consists of mainly 4 steps as shown in the Fig. 1. The first step is the image pre-processing. Once the image of the weld is captured, several image processing techniques are used pre-process the image. The output of this step is the all probable weld seam points. Once the probable weld points are detected, a line growing algorithm is used extract only the points which are actually on the weld seam. The third step is to select a best weld from the remaining weld points by analyzing the local neighborhood of the weld points. As the final step, the selected best point is grown along the weld joint to get the final weld joint. These steps are discussed in detail in following session.

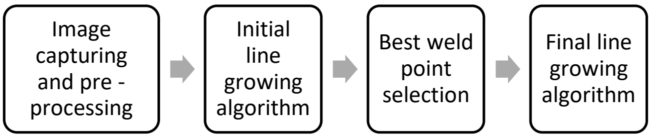

Fig. 1 Overview of the weld detection algorithm

A. Image capturing and pre-processing

A stereo depth camera is used as the vision sensor to detect and locate the weld path using the weld detection algorithm. The camera is mounted on a fixed link above the workspace with a stand-off distance of 50 centimetres. The image of the workspace before placing the workpiece in the workspace is prestored for the background subtraction (Fig. 2a). Once the workpiece is set up in the workspace, the image is captured manually (Fig. 2b). It is assumed that the entire weld joint is visible in the image captured.

The background image is subtracted from the captured weld image to extract the workpiece image. For further image processing, the subtracted image needs to be converted into greyscale image as show in the Fig. 3a. The initial weld points are identified using edge detection method. In this paper canny edge detection algorithm [14] is used for detecting the edges. It is clear from Fig. 3b that lot of faulty edges are detected in the image due to the surface texture, rust and scratches on the metal surface. As a first step of noise reduction, the length of all detected edges is calculated. Small detected edges with length of less than 50 pixels are not considered as weld edges and removed from the edge image (Fig. 3c). Further noise reduction is achieved by applying dilation and erosion followed by skeletonization. Dilation is a process which adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. When a noise image is dilated, small noises are combined together as shown in the Fig. 3d. The dilated image is then eroded (Fig. 3e) and Skeletonized to remove the noises. Skeletonization reduces the dimension of an object to generate its “skeleton”. The skeletonized image is shown in the Fig. 3f. These detected points are assumed to be the probable points on the weld. But as it can be seen from the image, there are still lot of points which are not actually on the weld joint. Those points which are not actually on the weld joint neighbourhood are eliminated using a line growing algorithm in the next step.

B. Initial line growing algorithm

To detect the actual points on the weld, the line growing algorithm suggested in [15] is used. The initial weld joint points are shown in Fig. 4. The line growing algorithm is based on the assumption that weld joint is darker than its surroundings due to the gap between two work pieces.

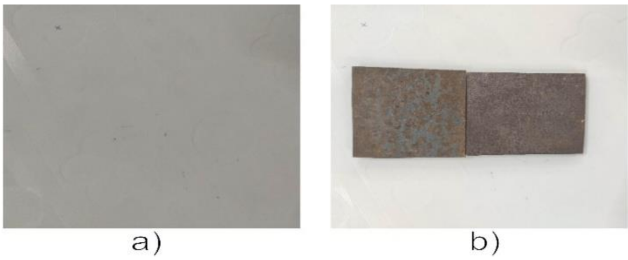

Fig. 2 (a) Background image (b) Captured weld image

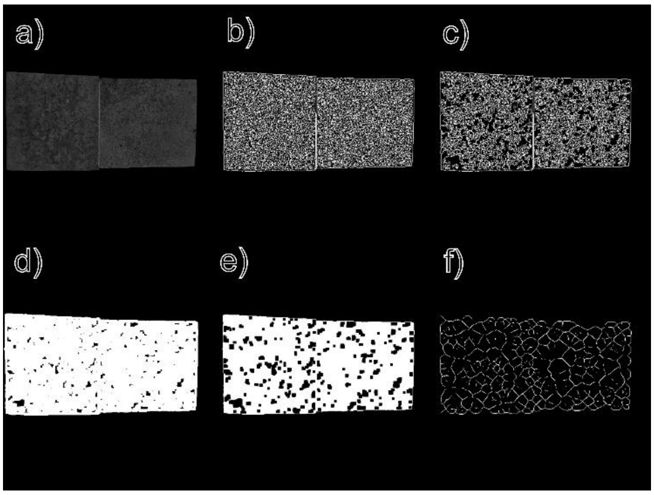

Fig. 3 (a) Grayscale image (b) Canny edge detection (c) Edge image after noise removal (d) Dilated image (e) Eroded image (f) Skeletonized image

Therefore, we can check if a given point is on weld joint or not by growing that point in the direction of “darkest” or “low intensity” pixels. If the point is on the weld joint, the line will grow through a smooth and consistent path. If the given point is not on the weld joint, the line will grow in an irregular manner since there is no path of consistently dark pixels to guide it.

The search and move directions for the line growing is shown in the Fig. 5. The aim of line growing algorithm is to grow the line with a single pixel width length in the direction of weld joint starting from initial weld points. At the current point, the point decides to move in one of the eight directions which have lowest pixel intensity. The pixel intensity of each direction is calculated using

$$D_{j}=\sum_{i=1}^{3} I(u_{ji},v_{ji}) \qquad (1)$$

Where Dj is the directional intensity of j^{th} direction, I is the grayscale intensity and (u_{ji},v_{ji}) is co-ordinate of the i^{th} pixel of the j^{th} direction. The next point to grow the initial weld point is chosen by finding which of the eight direction has the lowest total pixel intensity. To prevent the node growing backwards, one more condition is added. If the direction contains a previously visited node then that direction is no longer considered. This will prevent the line to grow along the path that it has already grown.

Fig. 4 Initial weld points

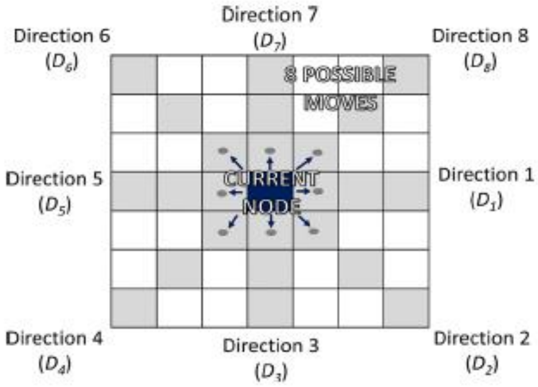

Fig. 5 Directional probes

In the initial line growing each initial weld point is grown over 100 pixels. As seen in the Fig. 6 the points which are not actually on the weld joint have produced irregular lines, while the actual weld joint points grown in a smooth line. In order to remove the points which have grown in erratic manner, directional threshold is used. The number of directions used out of 8 directions is recorded during the growth of the line. Any points used more than 2 directions in the line growth is no longer considered as weld point. Any point with directional threshold 2 or less is considered as weld point. Only the points grown in smooth manner is shown in Fig. 7a and the corresponding initial points are shown in red points in Fig 7b.

C. Selecting the best seed location

In Fig. 6 it can be seen that there are several points extracted as final weld points. For finding the final weld path, a best weld point which is closest to the centre of the weld joint has to be selected. It is done by analysing the grayscale pixel intensity around 3 x 3 neighbourhood of each final weld point. Points closer to the centre of the weld joint will have lower intensity pixels around it compared to other points which are on the edges of the weld joint. Therefore, the sum of neighbourhood pixel intensities for the points closer to the weld joint centre will be low. The point with least total pixel intensity in its neighbourhood is selected as the best weld point for the final line growing stage. The best point selected is shown with green marker in Fig. 8. As we can see, the selected best point is exactly aligned to the centre of the weld joint.

D. Final line growing algorithm

The final weld joint is detected using the line growing algorithm explained earlier in this paper with few modifications. In initial line growing algorithm, the line is grown over 100 pixels. Since the length of the weld joint is not known, the growth cannot be limited to a fixed length. The directional threshold is used to terminate the line growth. The directional threshold of 4 is used to check if the line has grown outside of the weld or not. Once line has reached end of the weld joint, it will no longer grow smoothly. The line will start growing in irregular directions outside the weld joint since there is no longer the path of low intensity to guide the line grow. When the number of directions used has reached more than 4, the growth of the line is terminated. Secondly, the line was grown only in a single direction in the initial line growing algorithm. Since the selected best weld point can be located anywhere in the weld joint, the final line growing algorithm is grown in two directions. Once the line has terminated from growing in first direction, it again starts from the best final weld point in the opposite direction so that the entire weld joint is detected. The result is shown in Fig. 9.

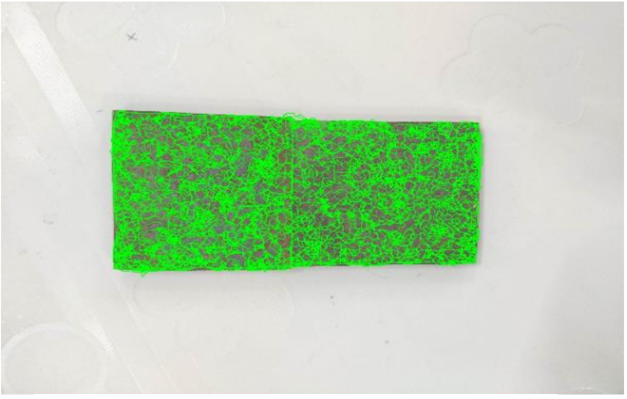

Fig. 6 Initial line growing result

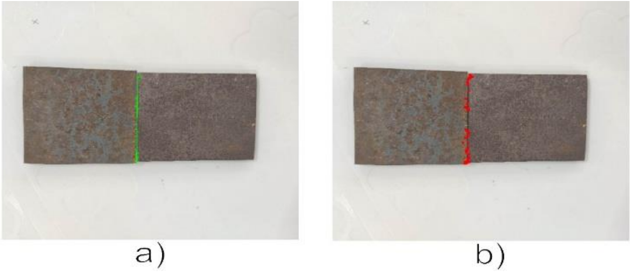

Fig. 7 a) initial line growing results after the direction change threshold b) Final weld joint points

As it can be seen from Fig.9, the line has grown outside of the weld joints at both ends. This is due to higher directional threshold of 4 is used in the final line growing algorithm to ensure full weld joint is detected. Hence both the ends of the grown weld must be trimmed to get the perfect weld joint. As discussed in the initial line growing algorithm, we can check if a given point is a weld joint point by growing that point along the lowest intensity path. The end point of the final detected line is grown over 100 pixels with a directional threshold of 2. If the end point is grown in an erratic manner, then that point is no longer considered as a weld point and it is removed from the final detected weld joint. This procedure is repeated until first successful weld joint is detected. This will remove all the points that has been grown out of the weld joint. The final detected weld joint after end trimming is shown in Fig. 10.

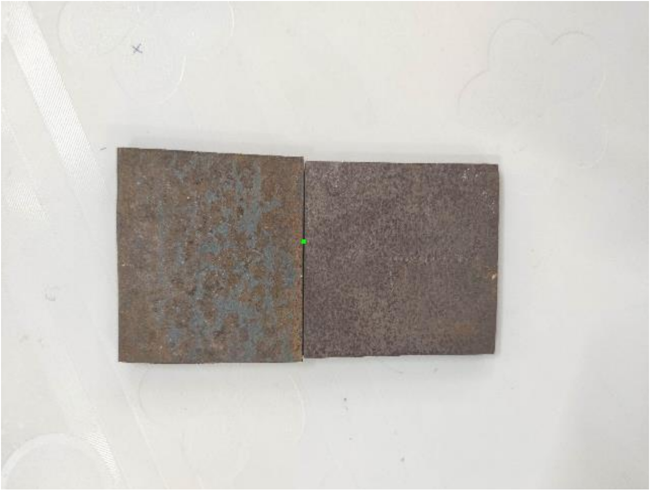

Fig. 8 Best seed location

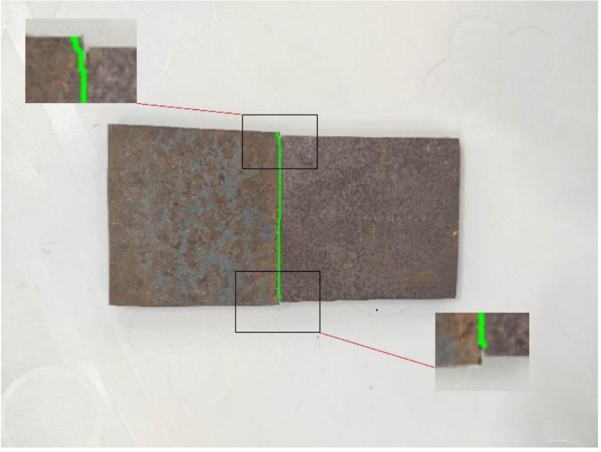

Fig. 9 Final weld joint before end trimming

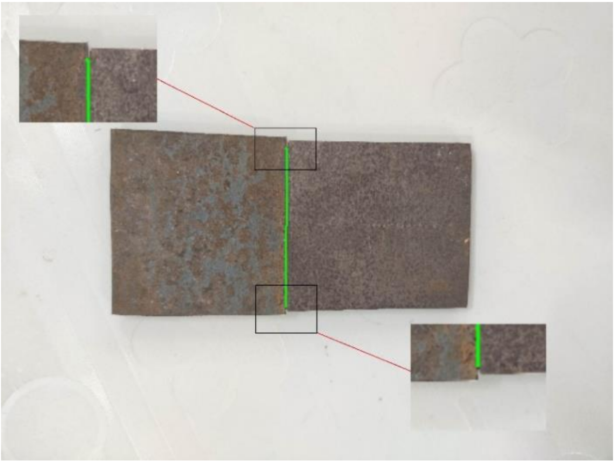

Fig. 10 Final weld joint after end trimming

III. RESULTS

In this session the test results of weld detection algorithm are discussed. Four different test cases are considered to test the weld detection algorithm. The error between the detected weld joint and actual weld joint is calculated to determine the accuracy of the algorithm.

A. *Weld detection results*

To ensure the algorithm is flexible to the orientation and shape of the butt joint, four different test cases are considered which are; Case A) straight butt joint, Case B) straight butt joint in different orientation than that of Case B, Case C) saw tooth butt joint and Case D) right angle butt joint. The different seam shapes are used to ensure that the algorithm is flexible to the shape of the weld joint. Case A and Case B have same weld shape, but the weld joint is placed in different orientations to ensure the orientation of the weld joint does not affect the weld detection algorithm.

Fig. 11 shows the original captured image and the result of the weld detection. As can be seen, the proposed method is capable to detect and identify the different weld path shapes and weld joints placed in different orientations. The detected weld joint is located centrally in the weld path line for all the four test cases.

B. *Butt welding joints matching result*

The start, end and auxiliary points are used to determine the accuracy of the detected weld joint. The pixel co-ordinates of the test points are taken from the images shown in the Fig.

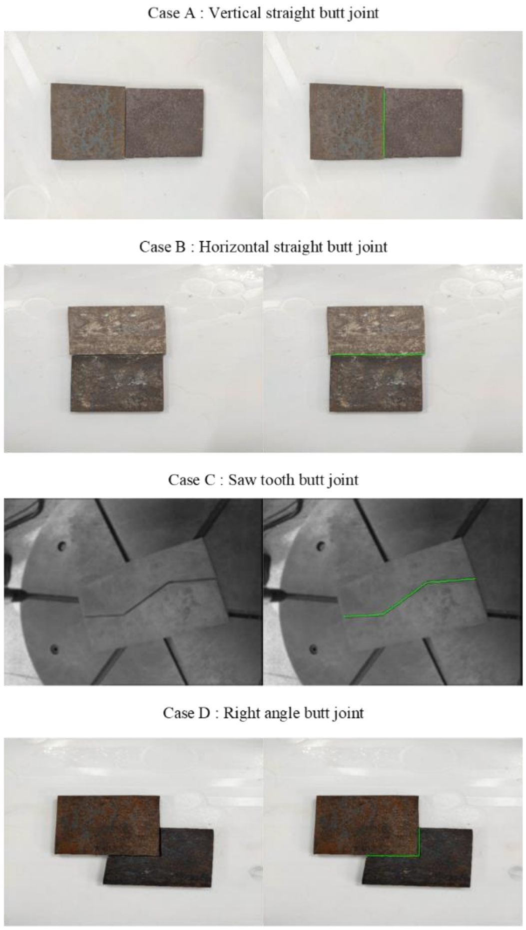

Fig. 11 Weld joint case studies (captured and detected image)

10. For Straight butt joints in Case A and Case B, the auxiliary points are selected at the midpoint of the weld joint. For saw tooth butt joint in Case C, the auxiliary points are considered at the two corner points of the saw tooth. For right angle joint in Case D, the auxiliary point is selected at the right-angle corner point. The image matching errors for all the four test cases is shown in Table 1. As seen in the table 1, the proposed method has a smaller error.

Table 1 Image matching errors

Test case	Test Points	Actual Position (x, y) pixel	Detected Position (x, y) pixel	Match Error
Case A	Start	[299,323]	[297,319]	[2,4]
	Midpoint	[299,234]	[297,234]	[2,0]
	End	[299,153]	[296,152]	[3,1]
Case B	Start	[166,230]	[171,230]	[-5,0]
	Midpoint	[279,230]	[279,228]	[0,2]
	End	[391,229]	[397,227]	[-6,2]
Case C	start	[195,302]	[200,304]	[-5, -2]
	Corner 1	[294,300]	[296,297]	[-2,3]
	Corner 2	[405,212]	[403,215]	[2, -3]
	End	[520,205]	[525,205]	[-5,0]
Case D	Start	[255,298]	[257,298]	[-2,0]
	Corner	[386,301]	[384,300]	[2,1]
	End	[384,230]	[384,229]	[0,1]

IV. Conclusions

In this paper, a new algorithm for detecting the butt-welding joints using vision system, that can be used for robotic welding has been introduced. By using this algorithm, it can automatically recognize and locate the butt weld joints. In this paper three different shapes of weld joints are tested. The output of the four test cases shows that the algorithm is capable of detecting different shaped weld joints. Also test cases A and B shows that the algorithm does not affected by the orientation of the weld joints. In all case studies, the highest matching error is for case study B where the match error for end point is 6 pixels in x direction and 2 pixels in y direction. Meanwhile the lowest matching error is shown for case study D which is 0 pixel in x direction and 1 pixel in y direction.

Acknowledgment

This work has been supported by Nodal Centre for Robotics and Artificial Intelligence (NCRAI), Government Engineering College, Thrissur, Kerala.

References

[1] P Kah, M Shrestha, E Hiltunen and J Martikainen, "Robotic arc welding sensors and programming in industrial applications," Int J Mech Mater Eng 10:13, 2015

[2] Doyoung Change, Donghoon son, Jungwoo Lee, Donghun Lee and Tae-wan Kim, "A new seam-tracking algorithm through characteristic-point detection for a portable welding robot," Robotics and computed-integrated manufacturing, vol. 28, pp. 1–13, 2012

[3] Amruta Rout, B.B.V.L Deepak and B.B. Biswal, "Advances in weld seam tracking techniques for robotic welding: A review", Robotics and computed-intergrated manufacturing, vol. 56, pp 12 – 37, 2019

[4] M. Kong, F.H Shi, S.B Chen and T Lin, "Recognition of the initial position of weld based on the corener detection for welding robot in global environment", Robot. Weld., Intellige. & Automation, Vol 362, pp 249 – 255, 2007

[5] X.Z. Chen, S.B. Chen and T. Lin, "Recognition of macroscopic seam for complex robotic welding environment", ", Robot. Weld., Intellige. & Automation, Vol 362, pp 171 – 178, 2007

[6] Mitchell Dinham and Gu Fang, "Autonomous weld seam identification and localisation using eye-in-hand stereo vision for robotic arc welding", Robotics and computed-intergrated manufacturing, vol. 29, pp 288 – 301, 2013

[7] Kevin Micallef, Gu Fang and Mitchell Dinham, "Automatic seam detection and path planning in robotic welding", Robot. Weld., Intellige. & Automation, Vol 88, pp 23 – 32, 2011

[8] Hairol Nizam Mohd Shah, Marizan Sulainman, Ahmad Zaki Shukor, Zalina Kamis and Azhan Ab ahman, "Butt welding joints recognition and location identification by using local thresholding", Robotics and computed-intergrated manufacturing, vol. 51, pp 181 – 188, 2018

[9] Fanhuai Shi, Tao Lin and Shanben Ghen, "Efficient weld seam detection for robotic welding based on local image processing", Industrial Robot: An international journal, vol. 36, pp 277 – 283, 2009

[10] S.B. Chen, X.Z. Chen and T. Qiu, "Acquisition of weld seam dimensional position information for arc welding robot based on vision computing", Journal of intelligent and robotic systems, vol. 43, pp 77 – 97, 2005

[11] L. Zhou, T. Lin and S.B. Chen, "Autonomous acquisiton of sem coordinates for arc welding robot based on visual servoing", J Intell Robot Syst, vol. 47, pp 239 – 255, 2006

[12] Yanling Xu, Na Lv, Gu Fang, Shaofeng Du, Wenjun Zhao and Zhen ye, "Welding seam tracking in robotic gas metal arc welding", Journal of materials processing tech., vol. 248, pp 18 – 30, 2017

[13] Hairol Nizam Mohd Shah, Marizan Sulainman, Ahmad Zaki Shukor,Muhammad Herman Jamaluddin and Mohd Zamzuri Ab Rashid, "A review paper on vision based identification, detection and tracking of weld seams path in welding robot environment", Modern applied science, vol. 10, Hairol Nizam Mohd Shah, Marizan Sulainman, Ahmad Zaki Shukor, pp 83 – 89, 2016

[14] Weibin Rong, Zhanjing Li, Wei Zhang and Lining Sun, "An improved canny edge detection algorithm", International Conference on Mechatronics and Automation, 2014

[15] Mitchel Dinham and Gu Fang, "Detection of fillet weld joints using an adaptive line growing algorithm for robotc arc welding", Robotics and computed-intergrated manufacturing, vol. 30, pp 229 – 243, 2014