

Technical University of Denmark



Passive Visual Sensing in Automatic Arc Welding

Liu, Jinchao; Fan, Zhun

Publication date:
2011

[Link back to DTU Orbit](#)

Citation (APA):

Liu, J., & Fan, Z. (2011). Passive Visual Sensing in Automatic Arc Welding. Kgs. Lyngby, Denmark: Technical University of Denmark (DTU).

DTU Library

Technical Information Center of Denmark

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Passive Visual Sensing in Automatic Arc Welding

Jinchao Liu

Kgs. Lyngby 2011
MAN-PHD-2011

Technical University of Denmark
Department of Management Engineering
Building 424, DK-2800, Kgs. Lyngby, Denmark
Phone +45 4525 4800
info@man.dtu.dk
www.man.dtu.dk

Summary

For decades much work has been devoted to the research and development of automatic arc welding systems. However, it has remained a challenging problem. Besides the very complex arc welding process itself, the lack of ability to precisely sense the welding process, including the seam geometry and the weld pool, has also prevented the realization of a closed-loop control system for many years, even though a variety of sensors have been developed.

Among all the sensor systems, visual sensors have the advantage of receiving visual information and have been drawn more and more attentions. Typical industrial solutions for seam detection such as using laser scanners suffer from several limitations. For instance, it must be positioned some distance ahead to the molten pool and may cause problem when dealing with shiny surfaces. Existing techniques for weld pool sensing mostly rely on auxiliary light sources and are difficult to be widely employed in industry.

Compared to human welders, existing sensor systems exhibit severe limitations as mentioned above. With the protection of only a welding shield glass, i.e., without any auxiliary illumination (passive), human welders can extract visual information on the weld pool and the nearby seam as the feedback to adjust the welding torch and/or welding parameters. It is an attractive idea from both academic and industrial point of view to develop a vision system without using any auxiliary light sources which can nevertheless extract relevant information. However, interpreting the images captured in a passive way during welding is challenging and may heavily rely on sophisticated image analysis and machine learning techniques. This industrial PhD project has been founded to tackle this problem.

For the last three years, we have explored different possibilities and thoroughly investigated the development of a passive vision system which is only equipped with a single off-the-shelf CCD camera and optical filters, yet capable of extracting sufficient information for the control purpose.

From the hardware side, we have studied the selection of proper optical filters to reduce the interference of the extremely strong arc light and controlling the exposure time of the camera on the fly to capture different images for seam tracking and weld pool sensing.

From the software side, we have designed a passive seam detection algorithm based on robust estimation techniques which can detect the seam geometry very close to the weld pool region. For the weld pool boundary tracking, we have proposed three approaches based on deformable models. The first approach employs inflating balloons and snakes, which are two types of deformable models, to capture the weld pool boundary. The first approach relies on a special periodical initialization scheme and only work in short-circuit mode. In order to handle other modes of the arc welding process such as spray mode where the strong arc light exists continuously, we have proposed another two approaches in which the initialization does not rely on the short-circuit moment. The essence is that deformable models can be immune to spurious edges caused by strong arc light and/or reflection from the seam by incorporating prior information on regions and boundaries.

The main findings are organized and presented in this dissertation.

Resumé

Automatisering af svejsning har i årtier udgjort et intensivt forsknings- og udviklingsområde, men ikke desto mindre udestår der stadig mange krævende problemstillinger. Dette skyldes dels de meget komplekse svejseprocesser som er svære at modellere, men fraværet af egnede sensorer som f.eks. kan virke tæt på smeltebadet og den intense lysbue har yderligere gjort især opbygningen af løsninger baseret på lukket-sløjfe styresystemer endog meget vanskelig. Derfor har udviklingen af egnede sensorer været vigtig og mange sensortyper er blevet testet.

Blandt de mulige sensorprincipper har især de optiske sensorer tiltrukket sig opmærksomhed, fordi de på samme måde som en menneskelig svejser muliggør anvendelsen af visuel information. Kravene til blot fugefølgning på den ene side og til en direkte observation af smeltebadet på den anden er meget forskellige. I dag anvendes typisk såkaldte laser-skannere til automatiseret fugefølgning, men disse lider af adskillige begrænsninger. Eksempelvis skal de nødvendigvis virke et stykke foran svejsningens smeltebad ligesom de kan have problemer i forbindelse med blanke og derfor kraftigt reflekterende overflader. Til at observere selve smeltebadet anvendes i dag typisk systemer baserede på en ekstern kraftig lyskilde - ofte en laser - og disse er derfor vanskelige at bringe ud i industriel anvendelse.

Som diskuteret ovenfor har eksisterende sensorsystemer i sammenligning med en menneskelig svejser alvorlige begrænsninger. Med anvendelsen af et simpelt svejseglas kan en menneskelig svejser uden andre lyskilder på baggrund af visuelle observationer af smeltebadet og dets nære omgivelser uddrage nødvendige informationer til at kunne tilpasse svejsepistolens position og stilling såvel som at foretage en tilpasning af svejseparametrene. Det er derfor fra såvel et akademisk

som et industrielt synspunkt en tiltalende idé at udvikle visuelle sensorsystemer som uden anvendelse af en ekstern lyskilde (dvs. passivt) kan udtrække den relevante information for styring af svejseprocessen. Det er imidlertid meget udfordrende og krævende at fortolke billeder optaget på denne passive måde og en billedanalyse må derfor nødvendigvis i meget høj grad bygge på sofistikerede billedbehandlings og ”machine learning” teknikker. Nærværende erhvervs-PhD-projekt er etableret med det formål at arbejde med denne problemstilling.

Igennem de sidste tre år har vi udforsket forskellige muligheder og i detaljer forsket i udviklingen af passive vision baserede systemer, som kun er udstyret med enkle kommercielt tilgængelige kameraer samt passende optiske filtre, men alligevel er i stand til at uddrage den relevante og tilstrækkelige information om smeltebad og fuge til automatisk styring af svejsningen.

På hardware-siden har vi bl.a. studeret udvælgelsen af passende optiske filtre som kan reducere interferensen med den ekstremt intensive lys fra lysbuen, og samtidig har vi udforsket mulighederne for on-the-fly at ændre kameraets indstilling for på denne måde med samme kamera at kunne optage billeder, der kan anvendes for henholdsvis fugefølgning og smeltebads baseret styring.

På software-siden har vi udviklet en passivt fugefølgings-algoritme baseret på robuste løsninger som kan bestemme fugegeometrien meget tæt på smeltebadet. Med hensyn til smeltebads-geometri bestemmelsen har vi foreslået tre måder at udtrække smeltebads-geometrien på, som alle er baserede på billedbehandlingsteknikken ”deformable models”. Den første løsning anvender to typer af ”deformable models, nemlig ”inflating balloons” og ”snakes”, til at indfange smeltebads-geometrien. Den bygger desuden på en speciel periodevis gentagen initialisering som kun er mulig når lysbueulyset er svagt, og kan derfor kun anvendes ved kort-bue svejsning. For også at kunne håndtere de andre mulige svejseformer indenfor lysbuesvejsning såsom spray-bue svejsning, hvor lysbueulyset er intenst og vedvarende har vi foreslået endnu to løsninger, hvor initialiseringen ikke bygger på fraværet af lysbueulyset. Det er vist at ”deformable models” kan gøres immune overfor tilfældige forstyrrelser forårsaget af det stærke lysbueulyset såvel som over refleksioner i fugen ved at inddrage tidligere opnået information om regioner og geometri. Projektets hovedresultater er struktureret og fremlagt i nærværende afhandling.

Preface

This thesis was prepared at Engineering Design and Product Development, Management Engineering, the Technical University of Denmark in partial fulfillment of the requirements for acquiring the Ph.D. degree in engineering.

The Ph.D study has been carried out under the supervision of Associate Professor Zhun Fan from Management Engineering, Technical University of Denmark, Associate Professor Søren I. Olsen from Computer Science, University of Copenhagen, Dr. Jens K. Kristensen and Dr. Kim H. Christensen from Welding and Product Innovation, FORCE Technology in the period from June 2008 to September 2011.

The thesis provides an investigation of developing a real-time passive vision system functioning similar to the eyes of a well-trained welder, i.e., sensing the welding pool and detecting the seam simultaneously in a passive way. Feeding the visual information into the control system allows us to realize a passive seam tracker and closed-loop control of arc welding processes.

Lyngby, September 2011

Jinchao Liu

Acknowledgements

First and foremost, my special thanks go to my supervisors, Dr. Zhun Fan, Dr. Søren Ingvar Olsen, Dr. Kim Hardam Christensen and Dr. Jens Klæstrup Kristensen for leading me into this exciting industrial research project. I appreciate all their contributions of time, ideas to help me handle all the difficulties encountered during my entire PhD.

I would like to thank the colleagues in K&P group at DTU management for creating inspiring and friendly working environment.

I would like to express my thanks to the colleagues in the division of welding and production innovation at FORCE Technology for helping me in completing all the experimental works.

I would like to thank Dr. Yuan Li for fruitful discussions on visual servoing and helping me prepare for the first two real-time demonstrations.

Special acknowledgement is given to the committee: Prof. Mogens Blanke, Dr. Jon Spørring and Dr. Bill Lucas for agreeing to be the examiners of the thesis.

Last but not the least, I would like to thank my mother for giving me unconditional love, encouragement and support through all my life.

Contents

Summary	i
Resumé	iii
Preface	v
Acknowledgements	vii
Figurliste	xiii
Tabelliste	xx
1 Introduction and Related Work	1
1.1 Research Background and Related Work	1
1.2 Objectives and Scope of the Work	5
1.3 Contributions of the Dissertation	5
1.4 Dissertation Outline	6

2	A Passive Vision System for Arc Welding Automation: An Overview	9
2.1	Introduction	9
2.2	Hardware System	10
2.3	Software System	14
2.4	Summary	14
3	Seam Detection	17
3.1	Introduction & Previous Works	17
3.2	Image Acquisition	19
3.3	Investigation of Using Conventional Edge Detectors and Hough Transform	21
3.4	The Proposed Passive Seam Detection Algorithm	23
3.5	Experiments and Results	30
3.6	Summary	32
4	Weld Pool Visual Sensing	35
4.1	Introduction & Previous Works	35
4.2	Weld Pool Visual Sensing	36
4.3	Introduction to Deformable Models	41
4.4	Weld Pool Tracking Using Deformable Models	45
4.5	Approach I: Inflating Balloons	47
4.6	Approach II: Active Contours Driven by Region based Statistical Models	63

4.7	Approach III: Feature Selective AdaBoosting Geodesic Active Region	82
4.8	Summary	97
5	Towards Stabilizing Parametric Active Contours	101
5.1	Introduction	101
5.2	Related Work	103
5.3	A Complete Tangential Diffusion Force	104
5.4	Experiments	107
5.5	Summary	108
6	Vision-based Control	113
6.1	Introduction	113
6.2	Visual Servoing: A Brief Introduction	114
6.3	Visual Servoing for Seam Tracking	115
6.4	Experiments	121
6.5	Summary	122
7	Conclusions and Future Work	125
7.1	Contributions	126
7.2	Future Work	127
A	Parametric Active Contours Driven by Region-based Statistical Models	129
B	Setting the Ground Truth: Manual Segmentation of Weld Pools	133

C On Parametrization	137
D Textual Features	139

List of Figures

2.1	Experimental setup including a CCD camera with a housing, a manipulator and a welding machine.	11
2.2	The spectrums of the lights generated during welding where the magnitude of the weld pool light is negligibly small compare to the arc light. x axis represents the wavelength, y axis represents the magnitude. (a)The spectrum which was captured when the arc light was intensive. (b)The spectrum which was captured when the arc light was less intensive and close to the short-circuiting moment. From these two figures, we see that most energy of the arc light concentrates on the short wavelength part of the spectrum.	12
2.3	(a)The spectrums of the weld pool light generated during welding where the arc is totally off. Most energy of the weld pool light concentrates on the long wavelength part of the spectrum. It is therefore possible to separate these two different kinds of lights.(b)The spectrum of the lights generated during welding. Note that in this frame, both the arc light and the weld pool light can be observed. This again confirms that by some bandpass filters, we may reduce the intensity of arc light without blocking the weld pool.	13
2.4	The architecture of the software system in the proposed vision system.	15

2.5	GUI of the software system.	16
3.1	Steel plates with v-type seam. (a) with constant width. (b) with varying width.	18
3.2	Typical frames captured during welding in two experiments. . . .	20
3.3	A v-groove seam can be described by four edges, i.e., two outer edges and two inner edges.	20
3.4	An illustration of the ground truth of the seam in a frame. The outer edges of the v-groove are marked in green. The inner edges of the v-groove are marked in yellow. These four lines are what will be extracted from the images.	21
3.5	The gradient magnitude of a frame and the results of using Canny edge detectors. From these results, it is seen that the four lines are not well-defined by conventional edges such as the local optimums of the gradient magnitude.	22
3.6	Hough transform was applied to detect one outer edge. Note that the acceptable range of the parameter θ in Hough transform is specified as $70 < \theta < 89.5$ which constrain the possible orienta- tions of the detected lines. We mark the first four most significant detected line in cyan, yellow, green and blue. The true seam edge is plotted in red. It can be seen that the errors are significant. . .	23
3.7	A model which contains four straight lines for a v-groove seam. .	25
3.8	This figure illustrates the details of data preparation. Two image ROIs are calculated by computing the image moments. Candi- date points are then extracted within each ROI by searching the first and the last edge points along the vertical direction.	29
3.9	Seam detection results of the application I without torch oscilla- tion.	33
3.10	Seam detection results of the application II. Torch oscillation is necessary in this case.	34
4.1	A typical frame for weld pool sensing. The weld pool boundary is manually annotated here and marked in green.	36

4.2	Frames captured during welding. See the text for detailed explanation.	37
4.3	Applying Canny edge detector to the weld pool images. See the text for detailed explanation.	39
4.4	An explanation of different kinds of edges in frame 48 of the experiment 2 detected by Canny edge detector. Different edges are marked in different colors. It can be seen that the detected edges contain many spurious ones and noises. In this image, we also observe that one part of the weld pool boundary marked in yellow is missing and could not be detected by Canny edge detector.	39
4.5	Applying Otsu thresholding to the weld pool images. See the text for detailed explanation.	40
4.6	Applying thresholding to the weld pool images. The threshold was optimized for our application. See the text for detailed explanation.	40
4.7	An illustration of level set functions (Alexandrov, 2004). Instead of a 1D curve in parametric deformable models, a 2D surface here is maintained and manipulated in a geometrical deformable model. The topological change of a 1D curve can be easily achieved by manipulating the corresponding 2D surface. This is a great advantage of geometrical deformable models.	44
4.8	The frames are categorized into two groups: arc-on frames and arc-off frames. This takes advantage of the fact that the arc is generated or disappeared periodically in short-circuit arc welding.	49
4.9	This is a graphically illustration of categorizing frames into arc-off frames and arc-on frames. The arc-off frames are marked by circles. The curve shows the average intensity of all frames.	50
4.10	This figure illustrates the initialization of an inflating balloon in an arc-off frame. This is a binary image by thresholding the original frame. The green circle inside the white region represents the initial position of the inflating balloons.	50

4.11	The figure shows the time of processing arc-off and arc-on frames. The arc-off frames are marked in red diamonds. The arc-on frames are marked in green circles. The frames before the arc is generated are marked in black stars. The time of processing arc-off frames is approximately 80ms. For arc-on frames, it is 20ms.	55
4.12	Statistical results of testing Approach I in Experiment 1. The box plots are computed from 100 frames. See text for explanation. The definition of <i>error</i> can be found in Appendix B. It is worth noting that in the ground truth, the weld wire boundary is excluded deliberately. On the other hand, the algorithm does extract the wire boundary along with weld pool boundary. This gives rise to the large errors in the control points 17 ~ 20 in (b) which also corresponds to the outliers shown in (a). This applies to all the subsequent figures in this chapter when presenting statistical results of weld pool boundary tracking.	56
4.13	Statistical results of testing Approach I in Experiment 2. The box plots are computed from 20 frames. See text for explanation.	57
4.14	Statistical results of testing Approach I in Experiment 3. The box plots are computed from 50 frames. See text for explanation.	58
4.15	A zoomed in figure of Fig.4.18(h). The truth boundary of the weld pool is marked in yellow. The extracted boundary by the proposed algorithm is marked in green. The algorithm failed in this frame because there are more than one strong edges. Unfortunately, the algorithm did capture an edge, yet false weld pool boundary.	59
4.16	A zoomed in picture of Fig.4.19(d) where the tracking algorithm based on inflating balloons failed. It can be observed that the boundary of the weld pool in the red box is indistinct.	59
4.17	Experiment 1. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves.	60
4.18	Experiment 2. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves.	61
4.19	Experiment 3. Torch oscillation is required. The extracted weld pool boundary are marked in green curves.	62

4.20	(a)The original image. (b)The histogram. The histogram shows that the intensities in the arc light region are very high, and the intensities in the weld pool region are small.	67
4.21	The histograms of the background and the foreground in two frames.	68
4.22	The Euclidean distances between each two histograms are computed and presented as a distance matrix. The first 17 rows or columns are foreground histograms, the rest are background histograms. The similarity within one group and the dissimilarity between the groups can be easily observed in this distance matrix.	68
4.23	Weld pool boundary extraction using the Chan-Vese model. Iterations 1000. (a) Initialization with a large number of spoiled regions. (b) Final segmentation. (c) Initialization with a big circle. (d) Final segmentation. The results are unsatisfactory, see text for more explanation.	69
4.24	Modeling the weld pool region and the background using Gaussian Mixture Models which has two isotropic gaussian components. (a)Weld pool: $0.3054 * \mathcal{N}(254.45, 7.9^2) + 0.6946 * \mathcal{N}(81.5, 46.7^2)$. (b) Background: $0.8738 * \mathcal{N}(81.5, 46.7^2) + 0.1262 * \mathcal{N}(53.9, 31^2)$	71
4.25	The train errors are plotted again the iterations during optimization using EM.	72
4.26	The figure shows the time of processing frames. The mean processing time of the experiment 3 is 27.9ms.	74
4.27	An example of minimization of the potential energy during evolution. This shows that after around 200 iterations, the curve reached an equilibrium.	75
4.28	Statistical results of testing Approach II in Experiment 1. The box plots are computed from 100 frames. See text for explanation.	76
4.29	Statistical results of testing Approach II in Experiment 2. The box plots are computed from 20 frames. See text for explanation.	77
4.30	Statistical results of testing Approach II in Experiment 3. The box plots are computed from 50 frames. See text for explanation.	78

4.31	Experiment 1. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves. .	79
4.32	Experiment 2. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves. .	80
4.33	Experiment 3. Torch oscillation is required. Red circles indicate the initial positions of the active contours. The final curves are marked in green. Iterations 200.	81
4.34	All pixels are classified into four groups: $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4$. \mathcal{B}_1 represents the boundary points being between the weld pool region and the background, marked in green. \mathcal{B}_2 represents the boundary points being between the arc light region and the background, marked in blue. \mathcal{B}_3 represents the boundary points being between the weld pool region and the arc light region, marked in red. \mathcal{B}_4 represents the non-boundary points.	83
4.35	Crucial features selected by AdaBoost.	87
4.36	Boundary pixels in different classes described by the <i>intensity</i> and the 2 nd Haralick feature <i>contrast</i> . The black eclipses indicate the two-dimensional Gaussian distribution of each class learned from labeled data.	88
4.37	The performance of the strong classifiers against the number of the weak classifiers that were used to create the strong classifier. The x-axis represents the number of the weak classifiers. The y-axis represents the misclassification rate.	89
4.38	Testing the active contour driven by both boundary-based and region-based information on a synthetic image.	91
4.39	Statistical results of testing Approach III in Experiment 1. The box plots are computed from 100 frames. See text for explanation.	93
4.40	Statistical results of testing Approach III in Experiment 2. The box plots are computed from 20 frames. See text for explanation.	94
4.41	The results of using FSA-GAR on Experiment 1. The initial curve is marked in red. The final curve is marked in green. . . .	95
4.42	The results of using FSA-GAR on Experiment 2. The initial curve is marked in red. The final curve is marked in green. . . .	96

4.43	All extracted weld pools are shifted along x-axis and plotted in the image coordinate system. (a)Experiment 1 (b)Experiment 3 .	99
5.1	An illustration that parametric active contours can be numerical instable during evolution . (a)Control points bunched together. (b)Active contours failed.	102
5.2	Comparison of curve evolution with or without the proposed tangential force in a synthetic image.	109
5.3	Comparison of curve evolution with or without the proposed tangential force in an image of a plane. The red circles indicate the initial contours. The final curve are marked in green.	110
5.4	Comparison of the proposed complete tangential force and the previous incomplete tangential force in the plane image. The x-axis represents the iteration. The y-axis represents the standard deviation of the distance between neighboring points.	111
5.5	Comparison of the proposed complete tangential force and the previous incomplete tangential force in the weld pool image. The x-axis represents the iteration. The y-axis represents the standard deviation of the distance between neighboring points.	111
5.6	Comparison of curve evolution with or without the proposed tangential force in a weld pool image. The red circles indicate the initial contours. The final curve are marked in green.	112
6.1	Diagram of position-based vision servoing.	115
6.2	Diagram of image-based vision servoing.	115
6.3	The world, robot and camera reference frames.	116
6.4	The definition of the error of the robot following a straight line seam.	118
6.5	A comparison between the actual robot path and the true center line of the seam. The robot path is marked in blue. The true center line of the seam is marked in red. See text for more explanation.	122

6.6 The residual error of the robot path compared to the true center line of the seam. 123

6.7 A comparison of the welding results with and without feedback control by using the passive seam tracking system. It can be seen that the welding robot can follow the seam very well by making use of the passive seam tracking system. If, however, without using any seam tracking system, the offset is so significant and results in a failure. 123

B.1 An example of manually segmenting a weld pool boundary and representing all the points in a polar coordinate system(r, θ). The resolution of the polar angle θ is 7.2° , i.e., 50 rays. The polar axis is indicated in a red line segment. The rays are marked in green. The landmark-like points which are extracted manually and describe the weld pool boundary are marked in red circles. . 134

List of Tables

3.1	The corresponding elements of the first four detected lines in the accumulator array.	24
3.2	The specification of the seam and welding parameters used in the two different applications.	31
4.1	Parameters used in Approach I	52
4.2	Parameters used by Approach II	73
4.3	Parameters used by Approach III	90
4.4	Comparison of three weld pool boundary tracking approaches . .	97

CHAPTER 1

Introduction and Related Work

1.1 Research Background and Related Work

1.1.1 Arc Welding Automation

In modern industry, welding plays an important role for jointing materials and arc welding has been the dominant process for many years. The arc welding process is a group of techniques including gas tungsten arc welding (GTAW or TIG) and gas metal arc welding (GMAW or MAG). However, nowadays the majority of arc welding has still been made manually. The need of automating the arc welding process has become more and more intensive. There are a number of reasons. First, the productivity of the manual welding is low, the quality is varying significantly and the cost is high, which all made it very difficult to remain competitive in a global world. Second, the by-products generated during the welding process such as the airborne particles and gases, the high-intensity light and the heat are dangerous for human welders. Third, some special environments necessitate the usage of the robotic arc welding system, for examples, the environment exposed to a nuclear radiation or underwater etc.

Therefore, much attention has been paid to the development of automated arc

welding systems. However, the success has been limited. The reasons are twofold.

1. Arc welding processes are difficult to model due to the complex dynamic nature. Many attempts to build mathematical models of arc welding processes have exhibited their limitations such as computationally expensive and unreliable. Alternative techniques such as neural networks are therefore investigated by many researchers (Christensen, 2003).
2. There is still a lack of proper sensors which can extract sufficient information for closed-loop control. For decades, many different kinds of sensors have been developed such as the contact probe, temperature sensors, through-the-arc sensors, electromagnetic sensors, and sound sensors. However, they have their own limitations which motivate researchers to seek alternative techniques.

In our project, the focus has been to tackle the second item above, i.e., develop a proper sensor system to facilitate the arc welding automation. In general, in order to produce a high-quality weld, the system should have two main functionalities. First, it should have the capacity of directing the welding torch based on the detection of the joint location and geometry, namely “seam tracking”. Second, the system should be able to adjust the welding parameters based on the information of the weld pool as well as the joint, namely “adaptive control of the welding condition”. Both functionalities heavily rely on the sensor systems.

1.1.2 Vision Sensors in Automatic Arc Welding

Among all sensors which have been applied in arc welding automation, visual sensors possess the advantage of receiving the visual information and draw more and more attention in recent years. For instance, a typical industrial sensor solution for seam tracking is to use laser scanner. This permits us to have the accurate measurement of the joint location and geometry (Sicard and Levine, 1989; Kim et al., 1995; Christensen, 2003; Xu et al., 2008; Fridenfalk and Bolmsjö, 2003). Although laser scanners have been widely used for seam tracking, it still suffers from several limitations. First, it must be placed at a distance to the molten pool. Second, it may cause problems when dealing with shiny surfaces. Third, typically the laser scanner solution is only suitable for seam tracking, and not able to sense the weld pool.

Information of the weld pool is essential for controlling the welding process. Well-trained welders acquire visual information on weld pools as the feedback for

controlling the torch and other parameters as well. Many researchers have been motivated to develop instruments or sensors which can observe the weld pool and capture visual information. However, the interference of the high intensity arc light often prevents the acquisition of satisfactory images of the weld pool. To address this problem, the use of auxiliary light sources to illuminate the weld pool regions, hence suppress the arc light, has also been investigated (Abdullah et al., 2007, 2008; Houghton et al., 2011).

An alternative solution has also been reported in a series of papers (Song and Zhang, 2007a,b, 2008, 2009; Ma and Zhang, 2009; Wang et al., 2010, 2011; Ma and Zhang, 2011). In their works, a structured laser pattern is projected onto the weld pool at a certain angle and the reflected laser light is received by an image plane. A high-speed camera equipped with a bandpass filter (centered at laser wavelength) is employed to capture the reflected laser pattern. By using specially designed algorithms, the shape of the weld pool may be revealed. However, this solution is computationally expensive and also limited by the high cost of using high-speed cameras.

Using auxiliary light sources significantly simplifies the task of processing the images acquired by the vision system. However, it is preferable to not involve auxiliary light sources for safety reason in industrial environments. This is a motivation for many researchers to develop vision sensors without using any auxiliary illumination which are often referred to as *passive* vision systems (Brzakovic and Khani, 1991; Bae et al., 2002; Zhao et al., 2003; Wang et al., 2005; Zhang et al., 2006; Balfour et al., 2006; Shen et al., 2008; Shi et al., 2010).

In (Bae et al., 2002), Bae et al. developed a visual sensing system for automatic gas metal arc welding of the root pass of a steel pipe. The system is equipped with a CCD camera and optical lens to capture weld pool images. By triggering the camera in the short-circuit moment, the proposed system is able to obtain weld pool images without any interference of the arc light. This simplifies the task of extracting weld pool boundary. The limitation is that the vision system can only work in short-circuit mode of arc welding and loses all the information of weld pools when the arc light is presented.

In (Zhao et al., 2003), the authors proposed to reconstruct a three-dimensional shape of the weld pool from a single weld pool image in pulsed GTAW with wire filler by making use of an algorithm called shape from shading(SFS). The image acquisition system here is fully passive and without any auxiliary light resources. It is among the first attempts to reconstruct the three-dimensional shape of the weld pool, yet still far from being used in industry.

In (Wang et al., 2005), the authors presented a passive image sensing system to capture the images of both topside and backside of the weld pool in a TIG

(tungsten inert gas welding) welding process of aluminium alloy. They also proposed an approach based on edge detection and a back-propagation neural network to extract the weld pool boundary. After applying edge detection, the extracted edge image usually contain many false edges and noise. The authors showed that the selection of the true boundary can be done by making use of a back-propagation neural network. It is important to note that in this work, the authors argued that the weld pool is obtained by using arc illumination which is however sometime questionable as the boundary between the arc and the background may not be the true weld pool boundary.

In (Shi et al., 2010), the authors proposed to use a part-based modeling method to detect and locate the weld pool in gas tungsten arc welding. However, part-based modeling methods are usually designed for representing the visual appearance of objects composed of rigid parts arranged in a deformable configuration (Felzenszwalb and Huttenlocher, 2005) and therefore not suitable for capturing the 2D or 3D shape of a non-rigid object such as a weld pool.

To sum up, existing vision systems for seam detection and weld pool sensing exhibit limitations such as the involvement of auxiliary illumination. On the other hand, if we exclude auxiliary light sources, processing and interpreting images captured during welding become very challenging. We have shown that some existing passive solutions attempt to solve this problem by capturing “clean” and “neat” images, i.e., without the presence of strong arc light, at the moment of short-circuit. However, this means that they can only work in short-circuit arc welding. Another possibility is to employ sophisticated image analysis and machine learning techniques to process relatively “dirty” images, i.e., containing a certain amount of arc light and obtain useful information. However, there have been quite few papers investigating this problem so far.

It is worth noting that in most existing works, detecting and representing weld pools still rely on conventional edge detection techniques. Some attempted to employ more powerful tools such as part-based modeling techniques which is however not very suitable for representing weld pools. Some attempted to introduce some machine learning techniques such as neural networks to facilitate edge detection as a post-processing step. However, as we will present in this dissertation, edge detection itself may be problematic in detecting weld pools. Therefore, it is also one of principle concerns in our work to seek elegant and effective image analysis tools for detecting and representing weld pools.

1.2 Objectives and Scope of the Work

In this PhD project, the focus is to thoroughly investigate the possibility of employing image analysis and machine learning techniques to extract visual information of the weld pool and the nearby seam from the images captured during welding in a passive way.

The overall aim is to develop a passive vision system which can provide information for both seam tracking and adaptive control of the welding process. The system is only equipped with a single off-the-shelf CCD camera and a carefully-selected narrow bandpass filter, yet capable of detecting the seam geometry and the weld pool boundary simultaneously in real time.

The particular welding process in this PhD project has been selected as gas metal arc welding (GMAW) which is very common in industry and maybe the most challenging one due to the intensive and dramatically-varying arc lights and spatters. The techniques, especially the image analysis and machine learning techniques developed in this project, are expected to be valid in other arc welding processes as well.

1.3 Contributions of the Dissertation

A major contribution of the dissertation is that we have developed a real-time passive vision system which is only equipped with a single off-the-shelf CCD camera and a carefully-selected narrow bandpass filter, yet capable of detecting the seam geometry and the weld pool boundary simultaneously in real time. More specifically,

- Several Image analysis approaches have been proposed to interpret the images acquired in a passive way during welding. More importantly, we have shown that deformable models which show great advantages in incorporating prior knowledge into image segmentation are elegant and effective tools for detecting and representing different kinds of weld pools.
- We have proposed a new active contour model, named as *Feature Selective AdaBoosting Geodesic Active Region*. The essential idea is that a small amount of features can be selected, by means of the feature selective AdaBoost, from a much larger feature set to represent the pixels. In doing so, the speed of the active contour can be greatly increased.

- We have also proposed a new tangential diffusion term for stabilizing the evolution of parametric active contours. We have proved that this term is bounded and superior to the previous incomplete one in the existing literature.

More details can be found in Conclusion 7.

1.4 Dissertation Outline

This dissertation consists of seven chapters and four appendixes.

Chapter 1 presents a general introduction to this PhD project. The research topic in this study is formally introduced. The research background and related work are presented as well.

Chapter 2 presents an overview of the proposed passive vision system. The hardware system is presented in detail. In particular, selecting proper optical filters to suppress the extremely high intensity arc light is discussed. The overall architecture of the software system is also presented in this chapter and details are discussed in subsequent chapters.

Chapter 3 starts to present image analysis algorithms in the software system. A passive seam detection algorithm based on RANSAC is proposed to extract the geometry of the v-groove seam. The algorithm is capable of overcoming the severe interference of the arc light and measuring the seam which is close to the weld pool and arc light region without using any auxiliary light sources such as lasers.

Chapter 4 continues to present the software system by presenting three weld pool boundary tracking approaches. We show that to extract the complete boundary of the weld pool and track it from frame to frame, deformable models offer us elegant tools. In this chapter, three different approaches based on parametric deformable models for weld pool boundary tracking are presented. The first approach is based on inflating balloons and snakes (Kass et al., 1988). The second approach is based on active contours that are driven by statistical models. The third one called *Feature Selective AdaBoosting Geodesic Active Region* attempts to combine the first two approaches and achieves better performance. We evaluate these approaches on the weld pool images captured in three different welding applications, discuss their advantages and also point out their limitations.

Chapter 5 presents a new tangential diffusion term for stabilizing parametric active contours. Numerical instability often occurs in evolving of parametric active contours. This is mainly due to the undesired change of parametrization during evolution. In this chapter, we propose a new tangential diffusion term to compensate this undesired change. As a result, the parametrization will converge to a parametrization that is proportional to the natural parametrization which implies that the control points of the contour are uniformly distributed.

Chapter 6 discusses how to incorporate the visual information obtained by a vision system into the control system so that the control loop can be closed using the visual input. We have presented a case study of implementing a closed-loop seam tracker based on the proposed passive seam detection algorithm.

Chapter 7 concludes the dissertation by stating the major contributions of this dissertation and outlining possible directions of future work.

Appendix A shows the details of deriving the Euler-Lagrange equation of the parametric active contour driven by region-based statistical models.

Appendix B discusses manual segmentation of weld pool boundaries which serve as ground truths in evaluating a weld pool boundary tracking algorithm quantitatively in terms of image analysis results.

Appendix C presents the details of how the parametrization of a active contour is characterized.

Appendix D shows the texture features used in Feature Selective Adaboosting Geodesic Active Region.

CHAPTER 2

A Passive Vision System for Arc Welding Automation: An Overview

2.1 Introduction

Resulting from this industrial PhD project, a passive vision system has been proposed for visual information extraction in arc welding. The system is capable of extracting the geometry of the seam very close to the molten pool area, and simultaneously extracting the weld pool shape in real time. By using the extracted information, we will be able to close the control loop using the visual input. In this chapter, we shall give an overview of the proposed vision system from both hardware and software sides. Especially we shall discuss the selection of proper optical filters to suppress the high intensity arc light.

As discussed in Chapter 1, the purpose of developing a vision system is to extract sufficient information for closed-loop control of arc welding processes. There are two different kinds of tasks, namely *seam tracking* and *welding process control*. Both involve the information of the seam and the welding process which will be the geometry of the seam and the weld pool shape when visual sensors are used. The responsibility of a vision system is to obtain this information, i.e.

detecting the seam and sensing the weld pool. Moreover, we shall highlight several keywords of such a vision system:

1. *Sufficiency*. The extracted visual information should be sufficient for the control.
2. *Speed*. Since the vision system will be included in the control loop, it should be fast enough to capture and process sufficient frames.
3. *Low-cost*. Controlling costs is of great importance from the business point of view.
4. *Passiveness*. It is preferable to develop a passive vision system for the reason such as safety.
5. *Versatility*. It is preferable to use exactly the same hardware equipment to accomplish two different tasks.

In our study, we have developed a passive vision system which meets the above requirements. From the hardware side, we use a low-cost CCD camera and a carefully-selected narrow bandpass filter. The camera can be controlled on the fly to capture different frames for two different tasks so that two tasks can be accomplished with the same equipments. From the software side, several image analysis algorithms are proposed for seam detection and weld pool sensing.

The remainder of the chapter is as follows. First, Section 2.2 discusses the hardware system, especially the selection of proper optical filters and the active control of the camera to capture images for different purposes. In Section 2.3, we discuss the architecture of the software system. Finally, we summarize this chapter in section 2.4.

2.2 Hardware System

Fig.2.1 shows the experimental setup in our study, including a welding machine, a manipulator, the proposed vision system and a standard PC. The computer controls all the other three systems. The hardware part of the proposed vision system consists of a low-cost CCD camera and a narrow bandpass filter in front of the camera, both of which are protected by a housing.

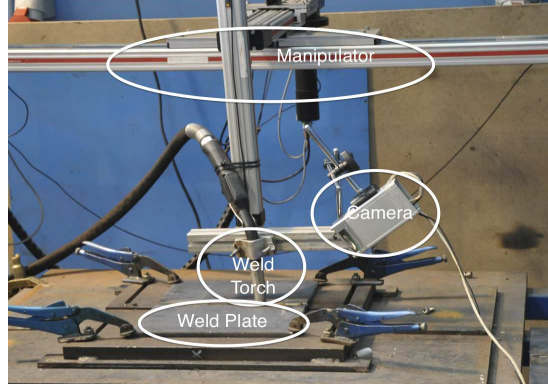
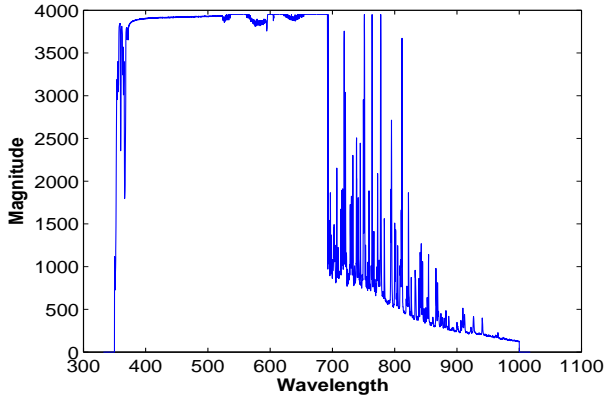


Figure 2.1: Experimental setup including a CCD camera with a housing, a manipulator and a welding machine.

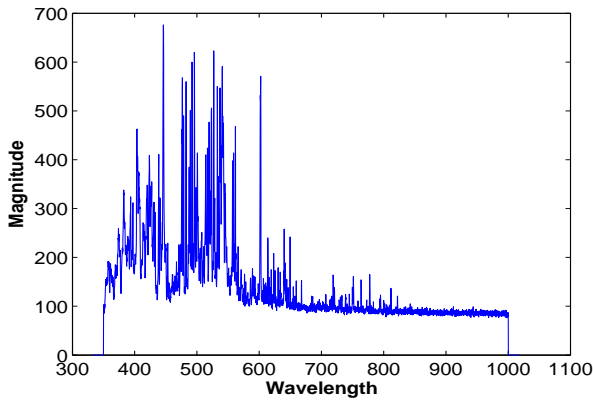
2.2.1 Optical Filters

For developing passive visual sensing in arc welding, a big challenge that one has to face is how to overcome the interference of the high intensity arc light and obtain the information that one needs. In our system, this problem is tackled by means of an optical filter and advanced image analysis methods. In this section, we show how to select a proper combination of optical filters to suppress the arc light and leave the detailed discussion of image analysis methods to the later chapters.

The selection of optical filters is based upon spectrum analysis. Fig.2.2 and Fig.2.3 show the spectrums of the lights which were captured by a spectrometer during welding. The distance from the optical probe of the spectrometer to the weld pool is approximately 10mm. Fig.2.2(a) shows the spectrum which was captured when the arc was intensive. The spectrums show that the region in the spectrum that is mainly composed of the weld pool light is separate from the region of the arc light, in spite of that the intensity of the arc light is very high. This motivates and permits us to use a narrow bandpass filter to decrease the intensity of the arc light and meanwhile increase the intensity of the weld pool light passing into the camera. In order to increase the signal-to-noise ratio, the center of the bandpass filter should locate within the interval of around 700nm to 1000nm. In this proposed system, a narrow bandpass filter with a center wavelength of $780 \pm 10\text{nm}$ was employed. In addition, a standard protection glass was also used to protect the bandpass filter from the spatters produced during welding.

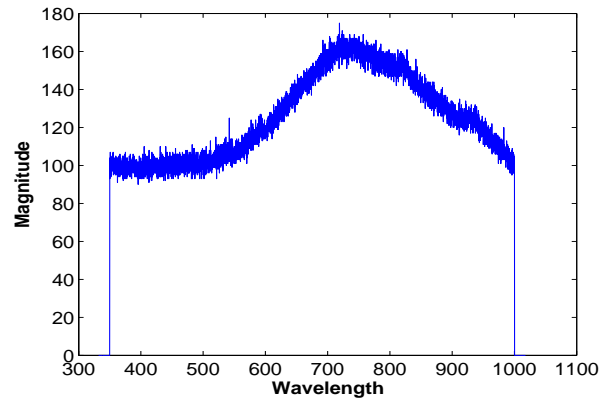


(a)

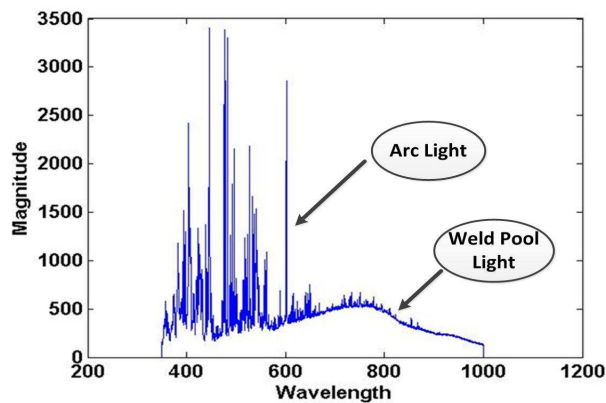


(b)

Figure 2.2: The spectrums of the lights generated during welding where the magnitude of the weld pool light is negligibly small compare to the arc light. x axis represents the wavelength, y axis represents the magnitude. (a)The spectrum which was captured when the arc light was intensive. (b)The spectrum which was captured when the arc light was less intensive and close to the short-circuiting moment. From these two figures, we see that most energy of the arc light concentrates on the short wavelength part of the spectrum.



(a)



(b)

Figure 2.3: (a)The spectrums of the weld pool light generated during welding where the arc is totally off. Most energy of the weld pool light concentrates on the long wavelength part of the spectrum. It is therefore possible to separate these two different kinds of lights.(b)The spectrum of the lights generated during welding. Note that in this frame, both the arc light and the weld pool light can be observed. This again confirms that by some bandpass filters, we may reduce the intensity of arc light without blocking the weld pool.

2.2.2 Image Acquisition

In our system, a camera control module is developed to control the exposure time of the camera on the fly. The reason is that seam detection and the weld pool sensing require different illumination conditions. In the case of seam tracking, we need some but not too much arc light to illuminate the seam. On the other hand, weld pool sensing requires that the arc light should be reduced as much as possible. Changing the exposure time on the fly, long exposure time for seam detection and short exposure time for weld pool sensing, can meet these two conflicted requirements.

2.3 Software System

This section will give an overview of the proposed software system. The details discussion will be presented in the subsequent chapters. The architecture of the software system is shown in Fig.2.4. The GUI interface is also shown in Fig.2.5. The overall software system is composed of two modules: a camera control module and an image analysis module. Section 2.2.2 shows the responsibility of the camera control module. The image analysis module contains all the image analysis algorithms and is responsible for extracting visual features from the images in real time. More specifically, seam detection algorithms and weld pool tracking algorithms are developed to extract seam and weld pool information respectively. In Chapter 3, we shall present a passive seam detection algorithm. In Chapter 4, three different approaches based on deformable models will be presented for weld pool boundary tracking.

2.4 Summary

In this chapter, an overview of the proposed vision system in this PhD project is given. From the hardware side, the system consists of a CCD camera and a narrow bandpass filter. We also discuss how to select proper narrow bandpass filters based on spectrum analysis. From the software side, several image analysis algorithms are proposed to extract corresponding visual information from the images.

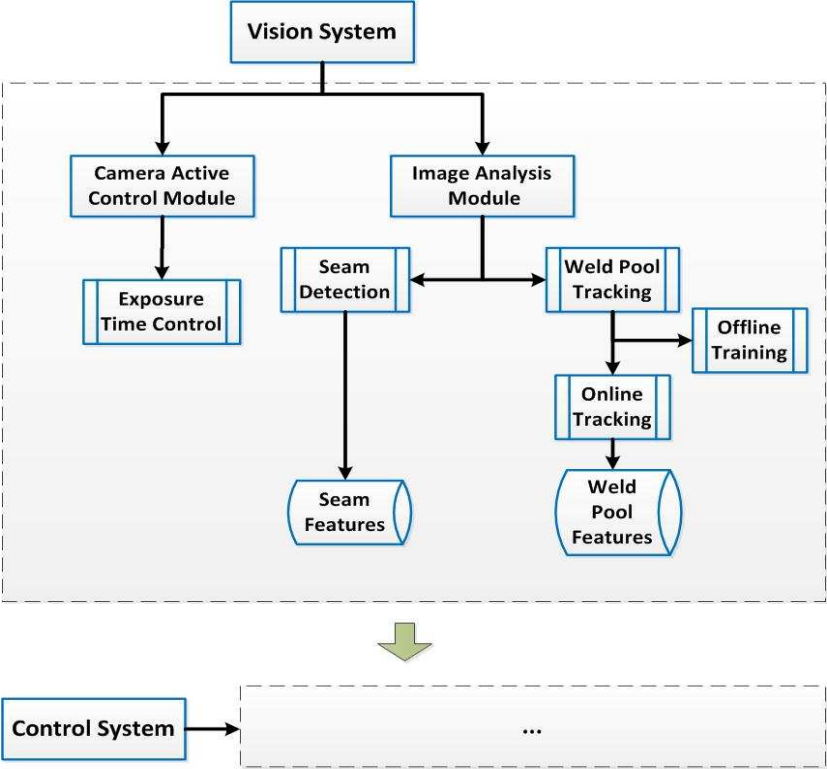


Figure 2.4: The architecture of the software system in the proposed vision system.

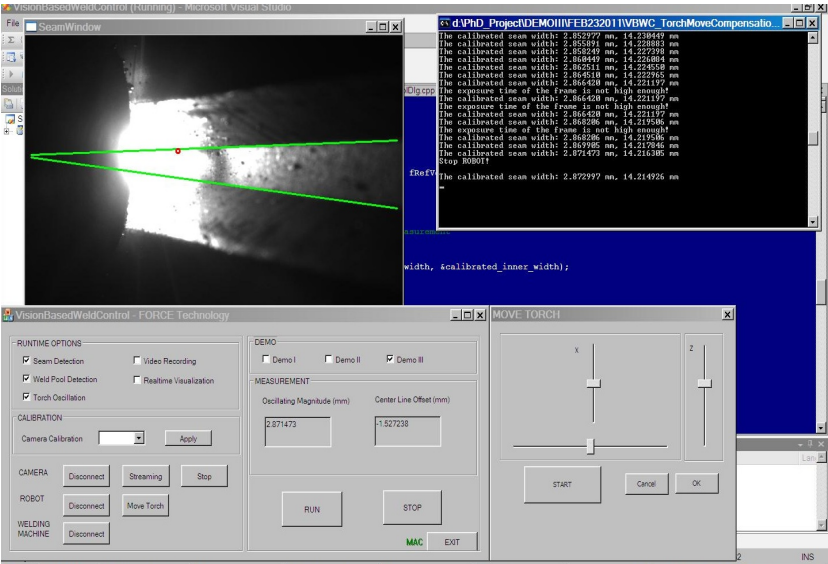


Figure 2.5: GUI of the software system.

Seam Detection

3.1 Introduction & Previous Works

Seam tracking is a term which we often use to refer to the task of navigating the torch and adjusting the welding parameters such as welding speed, welding voltage etc according to the seam information. It plays an important role in robotic arc welding. There are a variety of seams for different applications, such as V-type, T-type or I-type and so on. In our research, we are focused on v - groove which is one of the most widely used seam types to make edge-to-edge joints. For groove welding, the weld metal is melted and deposited inside the groove and penetrates and fused with the base material. Fig. 3.1 shows steel plates with v-type seam.

It should be noted that the seam shown in Fig. 3.1 was filmed under the natural light illumination only for the illustration purpose. In fact during welding, the visual detection of the seam profile is challenging due to the harsh environment, especially the interference of the high-intensity arc light.

Much effort has been made to address this problem from different perspectives. But one common strategy among all existing vision solutions is to employ optical filters to suppress the arc light, both for active and passive vision systems. For instance, the research in (Sicard and Levine, 1989) is concerned with the

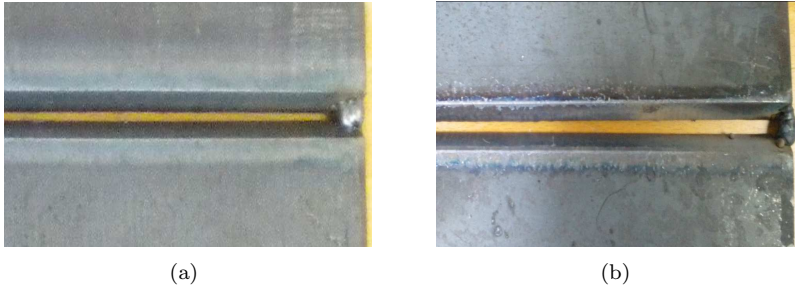


Figure 3.1: Steel plates with v-type seam. (a) with constant width. (b) with varying width.

development of an automatic seam tracking system. The author proposed to recognize the type and track the geometry of the seam by using laser scanner. More research using laser scanner can also be found in (Kim et al., 1995; Christensen, 2003; Xu et al., 2008; Fridenfalk and Bolmsjo, 2003). As a matter of fact, using laser scanner is currently a common technique for seam tracking in industry. Commercial products are provided by the companies such as FORCE Technology and Meta Vision Systems.

As discussed in Chapter 1, laser scanner solutions have some inherent limitations which have been motivating researchers to develop visual sensors for seam tracking without using any external illuminations. In (Shi et al., 2009), Shi et al. proposed an efficient weld seam detection algorithm which is able to handle different shapes of weld seams in butt joint welding. However, their approach is limited as it can only work before welding. In (Shen et al., 2008), Shen et al. developed a welding robot system in square-wave alternating current gas tungsten arc welding based on the real-time visual measurement. The system measures the offset of the torch to the seam center and the width of the seam gap and navigate the torch movement.

In this PhD project, the motivations of developing passive seam detection techniques are as follows. As we discussed in Introduction 1, the very popular laser scanner solutions face several difficulties which we aim at tackling by developing an alternative passive seam tracker. More importantly, coupled with the weld pool sensing techniques, we will be able to obtain visual information similar to what human welders can obtain during welding. This information is expected to be sufficient for developing a closed-loop control system. It is worth noting that for laser scanner solutions, the control objective is usually that the torch should follow the center line of the seam or a more complex path. By doing so, we hope that the weld pool can be positioned and shaped correctly. This

is in fact only a compromise since laser scanner solutions do not have the ability to sense the weld pool. It may however be problematic if for instance the welding wire bends. If the weld pool shape and the nearby seam geometry can be detected simultaneously, we will be able to realize a “true” closed-loop seam tracker where the control objective can be the “true” purpose, i.e., to position and shape the weld pool correctly.

Among all the existing works, most of them focus on either seam tracking or weld pool sensing. Very few of them consider detecting the seam and sensing the weld pool simultaneously with exactly the same camera setup. Very few approaches have the ability to detect the seam very close to the weld pool or tracking the weld pool robustly.

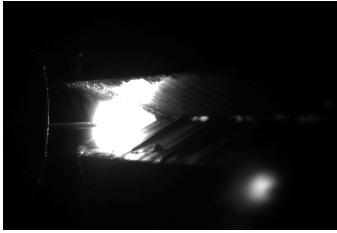
Seam detection in a passive way is challenging. As the seam does not emit any light, the only light source we may make use of to illuminate the seam is in fact the arc light. However, the intensity of arc light usually varies dramatically during welding. Moreover, the surface of the seam is often very rough and will create many spurious edges in the captured images. Extracting the seam from this cluttered background is usually difficult. In our work, we have been addressing this difficulty by making use of sophisticated image analysis techniques. Details will be explained in the subsequent sections.

The structure of this chapter is organized as follows. Beside the introduction to the background and previous works in the beginning, we precede by reporting the analysis of using state-of-the-art image analysis approaches particularly Hough transform associated with edge detectors for seam detection in Section 3.3. Section 3.4 presents a new passive seam detection approach. Section 3.5 presents the experiments and the seam detection results. Finally, we conclude this chapter in Section 3.6.

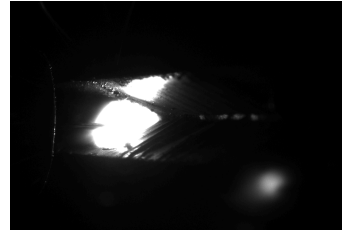
3.2 Image Acquisition

Typical frames captured by the proposed vision system are shown in Fig.3.2. These images are relatively good in the sense that they contain sufficient visual information of the seam for an experienced human welder to navigate the welding torch. By making use of advanced image analysis algorithms, it is possible to obtain the seam information which can be used by the control system for the welding torch navigation and welding parameters adjustment.

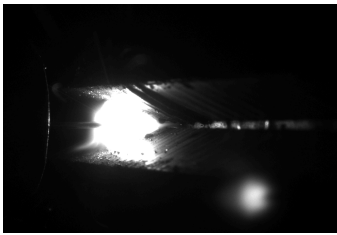
Prior to the development of a seam detection algorithm, we must define clearly what kinds of visual information is about to be extracted from the images. In



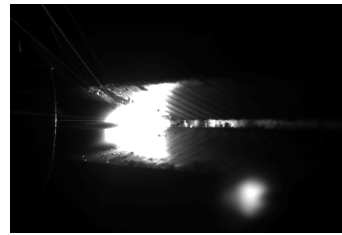
(a) Experiment 1, Frame 1



(b) Experiment 1, Frame 5



(c) Experiment 2, Frame 1



(d) Experiment 2, Frame 5

Figure 3.2: Typical frames captured during welding in two experiments.

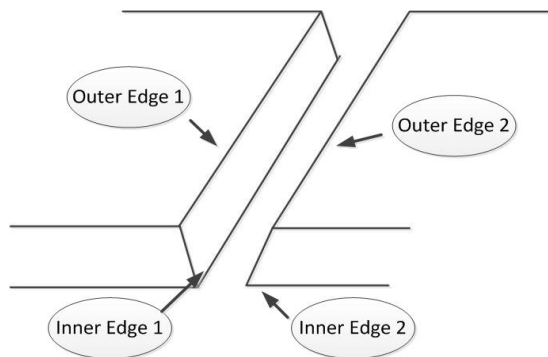


Figure 3.3: A v-groove seam can be described by four edges, i.e., two outer edges and two inner edges.

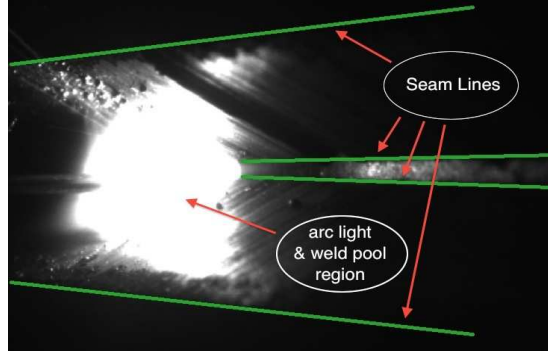


Figure 3.4: An illustration of the ground truth of the seam in a frame. The outer edges of the v-groove are marked in green. The inner edges of the v-groove are marked in yellow. These four lines are what will be extracted from the images.

general, the visual features that are crucial for seam tracking are basically the position, orientation and other geometrical quantities of the seam. We therefore propose to extract the four lines that describe the shape of the seam first and then calculate the geometrical quantities that we are interested in. Fig.3.3 gives an illustration. In Fig.3.4, we also mark the four lines that represent the seam in a real image for seam tracking.

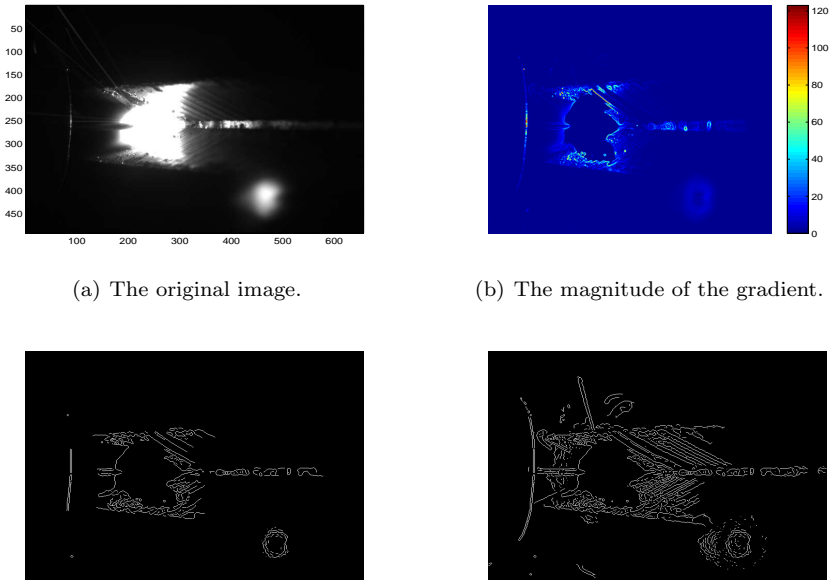
We have discussed in the beginning of this chapter that passive seam detection is challenging due to two reasons. One is that the arc light which serves as the only light source to illuminate the seam changes dramatically from frame to frame. The consequence is that the lines are not very well defined by conventional edges, such as gradient or textual edges. Another reason is that the surface of the seam is usually rough because of the cutting procedure. This will create many spurious edges in the images which makes the seam detection more difficult. In the next section, we shall investigate the possibility of applying conventional edge detectors and Hough transform to extract the seam lines.

3.3 Investigation of Using Conventional Edge Detectors and Hough Transform

A well-known technique of extracting lines in image processing community is to use edge detectors (Canny, 1986; Lindeberg, 1998) and Hough transform (Hough, 1959; Illingworth and Kittler, 1988; Zhang et al., 2008; Maji and Malik, 2009). In this subsection, we shall investigate the performances of this

conventional technique applied to the task of seam detection.

The detail is as follows. First, the Canny edge detector (Canny, 1986) is applied to the original image to obtain a binary edge image. Second, the Hough transform is applied to the binary image to detect the lines. In this investigation, MATLAB functions *edge*, *hough*, *houghpeaks* and *houghlines* were used. The threshold of the Canny edge detector was chosen automatically. The results of edge detection are presented in Fig.3.5. It is obvious that the seam lines are not well-defined by such as local optimums of the magnitude of the gradient.



(c) The result of Canny edge detector with a pre-defined threshold 0.1. (d) The result of Canny edge detector with the threshold selected automatically.

Figure 3.5: The gradient magnitude of a frame and the results of using Canny edge detectors. From these results, it is seen that the four lines are not well-defined by conventional edges such as the local optimums of the gradient magnitude.

Based upon the binary image, we may then apply Hough transform to extract potential lines. It is worth noting that the line in standard Hough transform is represented as

$$\rho = x * \cos(\theta) + y * \sin(\theta) \quad (3.1)$$

where θ represents the orientation of the line. Since in our case the seam edges are roughly horizontal, we can constrain the acceptable range of θ to boost the

performance of Hough transform. Fig.3.6 shows the results of detecting one outer line by using Hough transform. The four most significant detected lines are plotted in cyan, yellow, green and blue. The corresponding elements in the accumulator array is shown in Table 3.1. In Fig.3.6, the true seam edge is also plotted in red. It can be seen that the errors are significant and Hough transform yields unsatisfactory results.

In fact, the detected line marked in green is roughly the center line of the whole weld pool region. The cyan, yellow and blue lines are spurious edges caused by the rough surface of the seam. Using solely Hough transform can not solve this difficulty. Introducing prior knowledge and robust line fitting techniques into seam detection is essential. In our work, we propose to use a robust line detection technique based on RANdom SAMple Consensus(RANSAC) (Fischler and Bolles, 1981)..

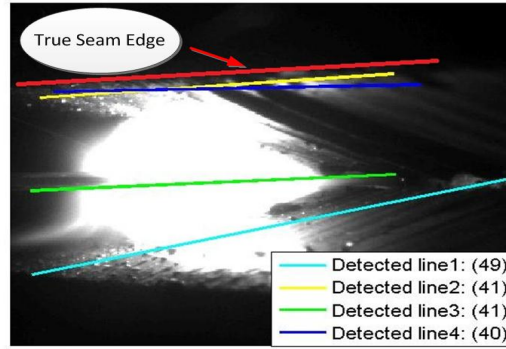


Figure 3.6: Hough transform was applied to detect one outer edge. Note that the acceptable range of the parameter θ in Hough transform is specified as $70 < \theta < 89.5$ which constrain the possible orientations of the detected lines. We mark the first four most significant detected line in cyan, yellow, green and blue. The true seam edge is plotted in red. It can be seen that the errors are significant.

3.4 The Proposed Passive Seam Detection Algorithm

As show in Fig.3.5, the seam images contain many spurious edges and noises. To tackle this, it is important to introduce prior knowledge into the seam detection.

Table 3.1: The corresponding elements of the first four detected lines in the accumulator array.

Detected Line	Accumulator
Line 1 in cyan :	49
Line 2 in yellow :	41
Line 3 in green :	41
Line 4 in blue :	40

In other words, we shall build a model which describes the desired visual features and carries prior knowledge and then estimate the model parameters according to the data. Based on this consideration, we proposed a seam detection approach which is capable of obtaining the position, orientation and other geometrical quantities of the seam in real time during welding.

The overall procedure is that we build a model with unknown parameters for the seam and then estimate these parameters according to the image data captured during welding. The quantities such as position, orientation etc will be computed according to the seam model afterwards.

3.4.1 The Seam Models

As shown in Fig.3.1, v-groove seams with constant or varying width can be described by four straight lines. A proper model in image domain for this kind of seams is thus four straight lines with certain constraints as illustrated in Eq.3.2 and Fig.3.7.

$$\begin{aligned}
 &y = k_i x + b_i \\
 &\text{subject to } \Phi(k_i, b_i) = 0, i = 1, \dots, 4
 \end{aligned} \tag{3.2}$$

Here we use $\Phi(\cdot) = 0$ to represent the constraints that are imposed on the model. Generally speaking, there are different kinds of constraints that may be useful, for example the ones that are due to the requirement of the welding techniques, or the ones that come from the physical restriction of the system. In the next few paragraphs we will present a detailed analysis of what kinds of constraints are particularly interesting in our application.

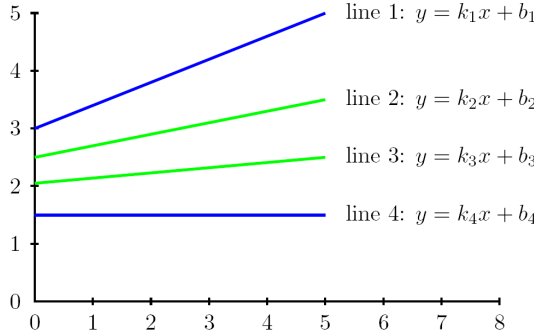


Figure 3.7: A model which contains four straight lines for a v-groove seam.

Physical Constraints

Our first concern is whether the lines are parallel. In the case of v-groove with constant width and depth, all four lines are physically parallel and ought to meet at the same vanishing points $(x_W^0, y_W^0, 0)$ which is represented by homogeneous coordinates in a world coordinate. In the case of v-groove with varying width and constant depth, the four lines that describe the seam will not intersect at the same vanishing point since they are no longer parallel. On the other hand, since the depth of the seam is constant, line1 and line2 should be parallel, so are line3 and line4. This fact allows us to use two vanishing points to constrain the model.

It is worth noting that the above physical constraints are valid and described in the world coordinate. In order to obtain the constraints in the image domain, the camera imaging and camera movement must be taken into account. In other words, we must verify whether the physical constraints are invariant to the transformation from the world coordinate to the image coordinate and how they change if not. We point out that two factors may affect these physical constraints.

- The perspective distortion due to the camera imaging.
- The translation of the camera due to the movement of the torch during welding.

Now let us examine how these factors affect the vanishing point constraints. In fact, the perspective transformation will bring the vanishing point from the

infinity $(x_W^0, y_W^0, 0)$ in the world coordinate to the point (x_I^0, y_I^0, z_I^0) in the image coordinate. More importantly, this vanishing point is invariant under the translation of the camera and therefore keeps unchanged for the entire image sequence.

- In the case of v-groove with constant width, the seam model with the vanishing point constraint can be expressed as follows.

$$y = k_i(x - \tilde{x}_0) + \tilde{y}_0 \quad i = 1, 2, 3, 4.$$

where $(\tilde{x}_0 = \frac{x_I^0}{z_I^0}, \tilde{y}_0 = \frac{y_I^0}{z_I^0})$ is the vanishing point where all the four lines meet. The degree of freedom (DOF) of this constrained model is 6.

- In the case of v-groove with varying width, the seam model can be as follows:

$$y = k_i(x - \tilde{x}_0) + \tilde{y}_0 \quad i = 1, 2 \quad (3.3)$$

$$y = k_i(x - \tilde{x}_1) + \tilde{y}_1 \quad i = 3, 4 \quad (3.4)$$

where $(\tilde{x}_0 = \frac{x_I^0}{z_I^0}, \tilde{y}_0 = \frac{y_I^0}{z_I^0})$ is the vanishing point where line1 and line2 meet and $(\tilde{x}_1 = \frac{x_I^1}{z_I^1}, \tilde{y}_1 = \frac{y_I^1}{z_I^1})$ is the vanishing point where line3 and line4 meet.

Without Using Any Constraints

It is to be noted that it is also an option that no extra constraints are imposed on the model. The advantage is that it is more general and allow us to have the same model for different seams, for example the v-groove seam with constant or varying width. On the other hand, it also has the disadvantage of containing more parameters compared to the model with extra constraints and therefore requiring more data for parameters estimation.

3.4.2 Estimation of the Model Parameters

There are various methods for estimating the model parameters including least-square, robust estimation methods, such as RANdom SAMple Consensus, abbreviated to RANSAC (Fischler and Bolles, 1981) and its variants etc. Usually the selection of the estimation method depends on the characteristic of the data. For instance, if we are confident or assume that all data are inliers, particularly drawn from a Gaussian distribution, least-square will be the best choice. On

the other hand, if the data is corrupted by outliers, robust estimation methods should be chosen, as the least-square method may be seriously affected by outliers and fail.

Assume that the parameters to be optimized is \mathbf{m} . In general, the estimation of the model parameters can be formulated as a Maximum Likelihood Estimation(MLE) problem.

$$\mathbf{m}^* = \operatorname{argmax}_{\mathbf{m} \in \mathbb{R}^n} \left\{ p(\text{Data}|\text{Model}(\mathbf{m})) \right\} \quad (3.5)$$

$$= \operatorname{argmin}_{\mathbf{m} \in \mathbb{R}^n} \left\{ -\ln p(\text{Data}|\text{Model}(\mathbf{m})) \right\} \quad (3.6)$$

or a Maximum A Posteriori Probability (MAP) estimation.

$$\mathbf{m}^* = \operatorname{argmax}_{\mathbf{m} \in \mathbb{R}^n} \left\{ p(\text{Model}(\mathbf{m})|\text{Data}) \right\} \quad (3.7)$$

$$= \operatorname{argmin}_{\mathbf{m} \in \mathbb{R}^n} \left\{ -\ln p(\text{Model}(\mathbf{m})|\text{Data}) \right\} \quad (3.8)$$

Maximum Likelihood Estimation: Least Squares

The data is given as four sets of points $(x_i^j, y_i^j), i = 1, \dots, 4, j = 1, \dots, N$. which are corresponding to four lines $y = k_i x + b_i, i = 1, \dots, 4$. Those points are detected in the image according to a certain criteria. Assume that the residual errors from the points to the potential line l_i are independently and zero-mean Gaussian distributed, i.e.,

$$(k_i x_i^j + b_i - y_i^j) \sim \mathcal{N}(0, \sigma_i) \quad (3.9)$$

By using maximum likelihood , the line l_i is estimated as follows.

$$(k_i^*, b_i^*, \sigma_i^*) = \operatorname{argmin}_{k_i, b_i, \sigma_i} \left\{ -\ln p(\text{Data}|\text{Model}) \right\} \quad (3.10)$$

$$= \operatorname{argmin}_{k_i, b_i, \sigma_i} \left\{ -\ln \prod_j \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp -\frac{(k_i x_i^j + b_i - y_i^j)^2}{2\sigma_i^2} \right\} \quad (3.11)$$

$$= \operatorname{argmin}_{k_i, b_i, \sigma_i} \left\{ \sum_j (k_i x_i^j + b_i - y_i^j)^2 + N \ln \sigma_i \right\} \quad (3.12)$$

This can be solved iteratively by gradient descent. Note that in this formulation, line detection becomes a *linear regression* problem (Bishop, 2007). The performance of this estimation relies on how well the data points meet the Gaussian assumption. If the residual errors are not Gaussian distributed or even outliers are contained, it is conceivable that the performance will not be satisfactory. An improvement can be to introduce more sophisticated distributions or learn the exact distribution from the data points, and then derive the estimation based on this new distribution. However, learning an exact distribution from samples is not easy. In our work, we propose to use an alternative framework for robust estimation based on RANSAC. The resulting algorithm is straightforward and easy to implement.

Robust Estimation: RANSAC

As mentioned above, we propose to use a RANSAC based approach to detect the seam lines. We shall make a brief introduction to RANSAC here. It is a model estimation technique that can tolerate a large portion of outliers and first proposed by Fischler et al. (Fischler and Bolles, 1981). The pseudo code of RANSAC is shown in Algorithm 1.

input : A set of points $D = \{x_i, i = 0, \dots, N\}$

output: The model parameters

1. Select a minimal set of points S_i randomly required to determine the model parameters;
2. Estimate the model parameters according to S_i , compute the residuals and classify into inliers and outliers according to a pre-defined threshold ϵ ;
3. Grade S_i in terms of the amount of inliers ρ_i , i.e., how many points agree with this model with the tolerance ϵ ;
4. Stop if the ratio of the number of inliers over the number of all points $\frac{\rho_i}{N}$ exceeds a pre-defined threshold τ . Otherwise, return to step (1);
5. Select the minimal set of points with the highest score and re-estimate the parameters using all inliers;

Algorithm 1: Pseudo-code of RANSAC

3.4.3 Data Preparation

To be able to estimate the model, we must have both the hypothesis which is a model carrying prior knowledge and a data set obtained from the results of image processing, as shown in Eq.3.8. A high-quality data set greatly simplifies

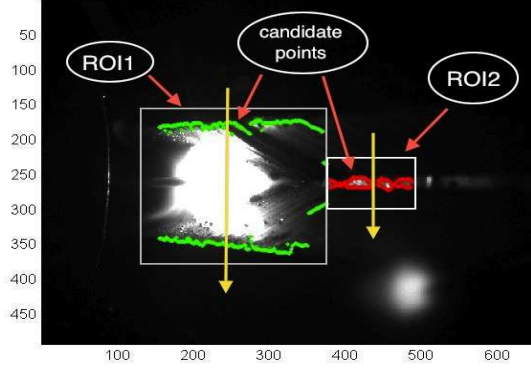


Figure 3.8: This figure illustrates the details of data preparation. Two image ROIs are calculated by computing the image moments. Candidate points are then extracted within each ROI by searching the first and the last edge points along the vertical direction.

the model, increases the accuracy of the estimation and reduces the computation time. Here we shall discuss the preparation of the data sets by image processing techniques.

Recall that the visual features(image features) for seam detection in this particular application are four lines that describe a seam. From the image processing point of view, these four lines are characterized as *edges* which are in general detected based on sharp, local changes in intensity. Previous discussions have illustrated that conventional low-level edge detection techniques failed in our application. Therefore we proposed to use a line search approach to detect candidate points for robust line fitting.

Calculation of the ROIs

In Fig.3.4, we show the ground true for one frame. Note that the desired outer lines and inner lines are in fact located at different regions in the image. For the sake of extracting different lines and reducing the processing time, we shall compute the ROIs in which the subsequent procedures are carried out. As shown in Fig.3.8, two rectangles will be specified by computing the image moments. ROI_1 covers the overall weld pool area and serves the purpose of extracting two outer lines of the seam. ROI_2 is placed just ahead the weld pool area for extracting the two inner lines of the seam.

Detection of the Candidate Points

Edge points for line fitting are computed via Canny edge detector. As discussed earlier, the line points here are not well defined by for example local gradient due to the dramatic change of illumination during welding. Therefore, we define the candidate points as the first or last edge points in the scan line which is along the vertical direction(depend on the camera setup). This column search is performed for each column of each ROI and produce a series of candidate points as shown in Fig. 3.8.

3.4.4 Pseudo-code

After obtaining the candidate points which may contain a number of outliers, RANSAC is then applied to fit the lines. The pseudo-code of the proposed seam detection algorithm is shown in Algorithm 2.

input : A frame $I_{m \times n}$
output: The four lines that describe a v-groove

1. Compute the image ROIs, i.e., position two rectangles;
2. Compute a binary edge image using Canny edge detector in two different image ROIs;
3. Detect candidate points which are defined as the first or the last edge points along the line perpendicular to the seam;
4. Estimate the model parameters based on the candidate points using RANSAC;

Algorithm 2: A Seam Detection Algorithm

3.5 Experiments and Results

Several experiments have been conducted to evaluate the proposed seam detection algorithms. In this section, we will present the results of applying the proposed seam detection algorithms to two different experiments. Both experiments are v-groove arc welding. The differences are summarized as follows:

- Application I: the inner width of the seam is small, around 1 mm. Torch oscillation is not necessary in the root path welding.

- Application II: the inner width of the seam is large and varies from 3mm to 5mm. Torch oscillation becomes necessary in the root path welding.

The detailed description of the seams and the welding parameters used in two different applications are given in Table 3.2.

Table 3.2: The specification of the seam and welding parameters used in the two different applications.

Application	I	II
Seam type:	v - groove	v - groove
Seam width:	narrow, constant	wide, varied
Workpiece	structured steel	structured steel
Torch oscillation:	no	yes
Wire-feed speed:	5.5 m/min	2.5m/min
Wire diameter:	1.0 mm	1.0 mm
Reference Voltage :	17.5 V	14.5 V
Measured Current:	~ 139 A	~ 80 A
Shielding gas:	Argon(82%)/CO2(18%)	Argon(82%)/CO2(18%)

The results of processing the frames captured during welding by the proposed vision system are presented in Fig.3.9 and 3.10. Fig.3.9 shows the seam detection results of the application I. It can be seen that the four lines that describe the v-groove workpiece have successfully extracted. Two outer and inner edges of the v-groove are marked by red and green lines, respectively. For almost all frames, the extracted outer lines are rather accurate. The quality of the extracted inner lines, on the other hand, are less stable, for instance, see Fig.3.9(a), (b), (h). The reason is that in this case the seam is rather narrow, only few candidates points can be detected when the illumination condition is bad.

It is worth noting that extracting the four lines are not the final goal, but provide the geometry of the seam. For instance, for most cases, we are interested in the width of the outer edges or the inner edges, and the center line of the seam. A relatively high frame rate, in our application more than 10 Hz , allows us to run a filtering algorithm such as Kalman filters afterwards to stabilize these features. In this work, we have also implemented a closed-loop seam tracker using the proposed passive seam detection approach. The detailed discussion and welding results are presented in Chapter 6.

Fig.3.10 presents the seam detection results of application II where the torch was oscillated. All four lines were extracted accurately. In addition to the four lines

that describe the seam, the center line of the seam is also computed accordingly and plotted as a yellow line in the figures.

3.6 Summary

The online detection of the seam geometry is of great importance in automatic arc welding. Typical industrial solutions such as using laser scanners face several difficulties. For instance, it is not able to detect the seam very close to the weld pool. These motivate us to develop a passive seam detection approach. Moreover, coupled with the weld pool sensing techniques which will be presented in Chapter 4, we will be able to establish a closed-loop seam tracker which controls the weld pool position and shape directly.

This chapter presents a passive seam detection approach which can detect the seam geometry very close to the weld pool. We have shown that conventional techniques such as using edge detection and Hough transform yield unsatisfactory results due to the fact that a number of spurious edges are present in the seam image. To tackle this problem, we propose to use a robust estimation technique called RANSAC which is able to tolerate a large amount of noises and spurious edges and nevertheless find the true seam lines. We examine the proposed seam detection approach in two different experiments. The results verify the feasibility of the proposed approach.

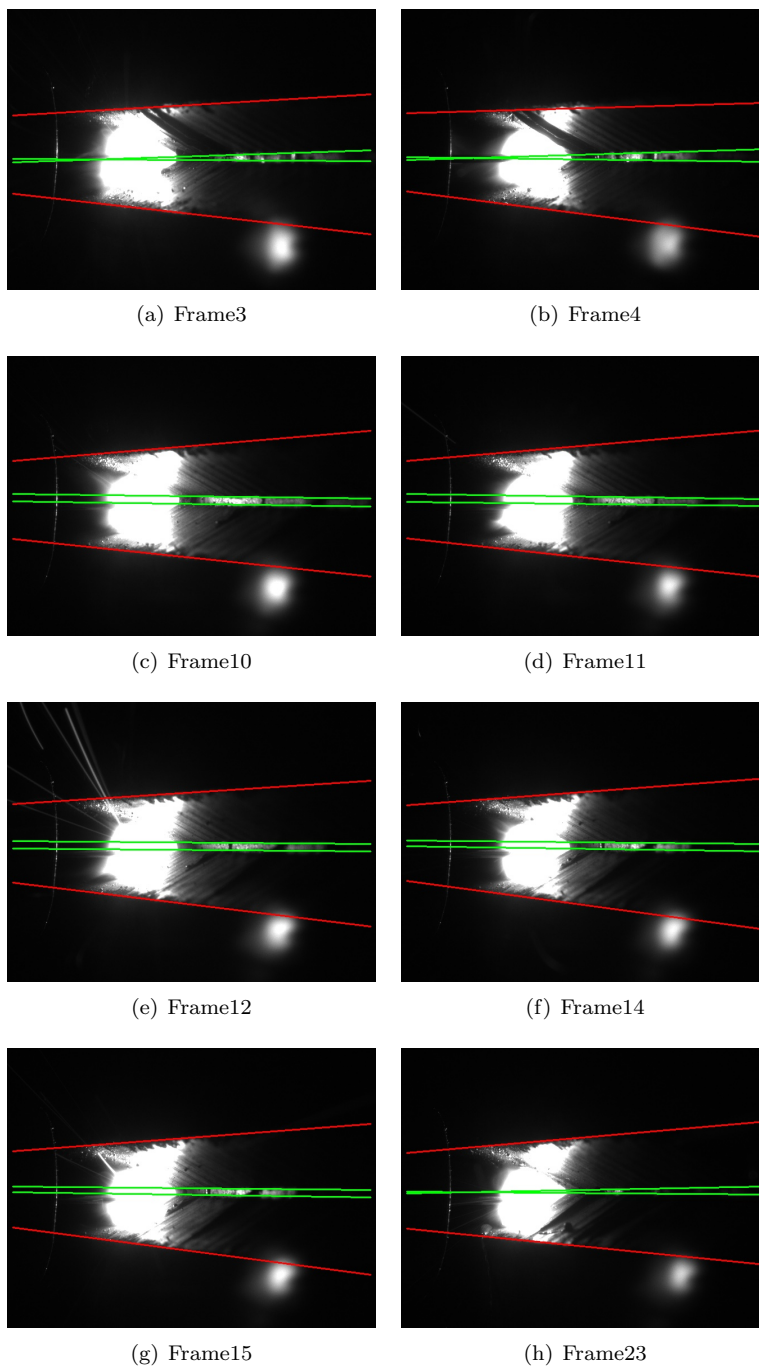


Figure 3.9: Seam detection results of the application I without torch oscillation.

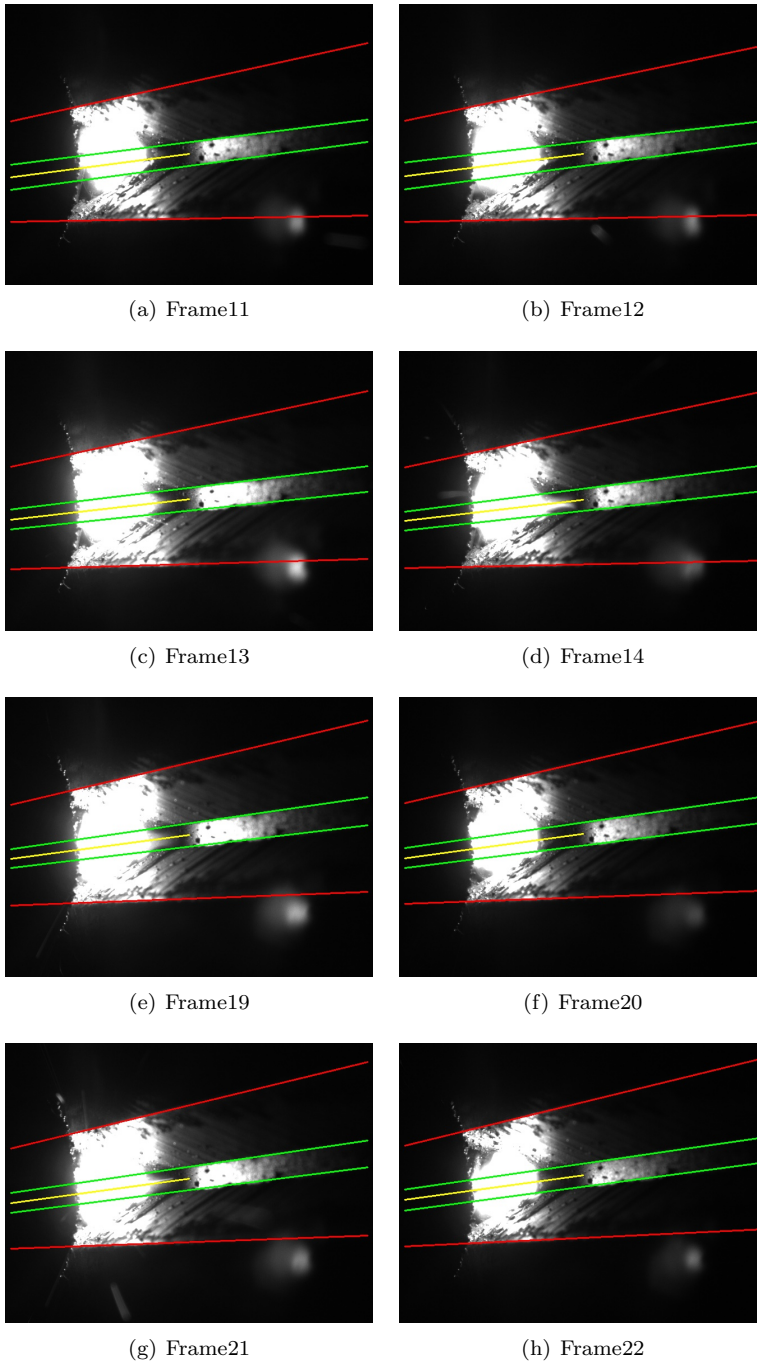


Figure 3.10: Seam detection results of the application II. Torch oscillation is necessary in this case.

CHAPTER 4

Weld Pool Visual Sensing

4.1 Introduction & Previous Works

Closed-loop control is of great importance in automatic arc welding for making a good welding such as good fusion, sufficient penetration. As discussed in Chapter 1, sensors that can obtain information reflecting the status of the weld pool region is one of the elements. In general, the visual information of the weld pool region contains the seam geometry and the weld pool shape. In Chapter 3, we have presented a passive seam detection algorithm which can extract the geometry of the seam very close to the weld pool. In this chapter, we continue to investigate image analysis techniques for extracting the weld pool shape.

For decades, much effort has been made to detect the weld pool shapes in a passive or active way. It has been discussed that our focus has been to detect the weld pool shape in a passive way. For a survey of existing works on passive weld pool visual sensing, please refer to Chapter 1, Section 1.1.2.

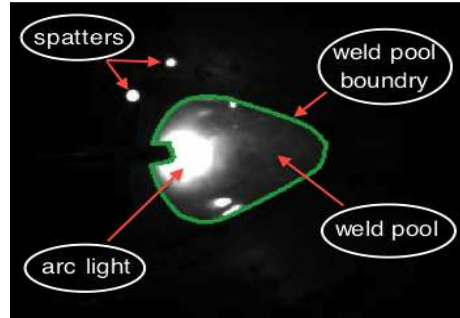


Figure 4.1: A typical frame for weld pool sensing. The weld pool boundary is manually annotated here and marked in green.

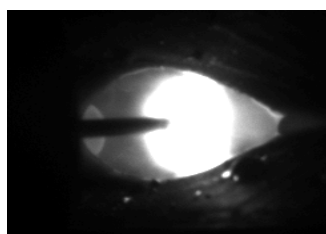
4.2 Weld Pool Visual Sensing

4.2.1 Image Acquisition

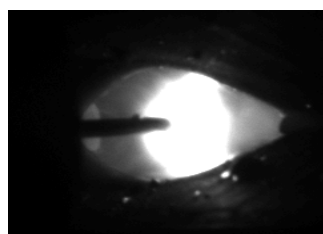
In Chapter 2, we have shown that relatively good weld pool images can be captured by using only a CCD camera and a narrow bandpass filter. No auxiliary illumination sources are needed. Fig.4.1 shows a typical frame of weld pool, as well as different regions in a typical weld pool image.

Frames captured from another two experiments are shown in Fig.4.2. Here several observations can be made:

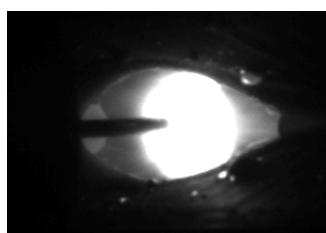
1. The weld pool boundary varies during welding and can be very different from one welding application to another. In fact, the 3D shape of a weld pool depends on many variables, including the geometry of the seam, the welding parameters, etc. Furthermore, what is captured in the image plane is the projected 2D shape transformed from the 3D weld pool and therefore will also be affected by the camera setup.
2. The weld pool region is partially covered by the very strong arc light. This is due to the nature of arc welding processes. In gas metal arc welding (GMAW), when the arc is generated, periodically in short-circuit mode and continuously in spray mode, the wire is melted, drops and forms the weld pool. Therefore the arc light appears above the weld pool region for most of the time.
3. The location and area of the arc light region can change during welding. The location is actually where the arc light is generated. This depends on



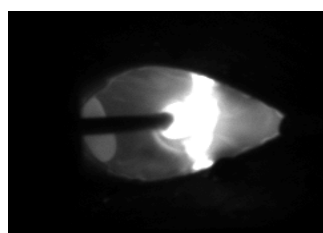
(a) Experiment 1, Frame 53



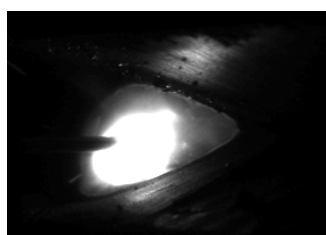
(b) Experiment 1, Frame 55



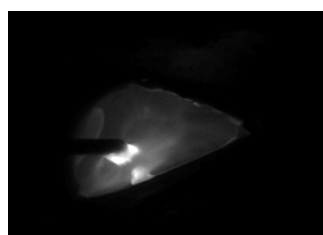
(c) Experiment 1, Frame 56



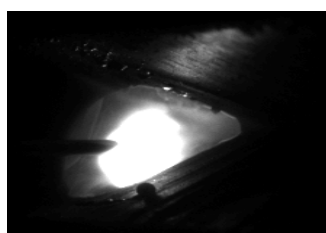
(d) Experiment 1, Frame 66



(e) Experiment 2, Frame 48



(f) Experiment 2, Frame 52



(g) Experiment 2, Frame 65



(h) Experiment 2, Frame 81

Figure 4.2: Frames captured during welding. See the text for detailed explanation.

the relative position of the welding torch and the workpiece. This can be observed easily in Fig.4.2 where weld pool images captured in two different experiments are presented, see Section 4.4.1 for detailed explanation of the experiments. In experiment 1, the welding torch is positioned in the center line of the seam. The arc is generated at both sides of the seam and we observe an almost symmetrical weld pool. The location of the arc is also in the center of the seam. On the other hand, in experiment 2, the welding torch is a little offset. The arc is generated mainly at one side of the seam and therefore mainly located in this side. The weld pool is no longer symmetrical.

What we are interested in and may benefit from is the shape of the weld pool. In the case of 2D, it is the boundary of the weld pool. We shall propose image analysis techniques to detect the weld pool boundary from the images in real time. In the next subsection, we shall present the investigation of using conventional image processing techniques for weld pool boundary tracking.

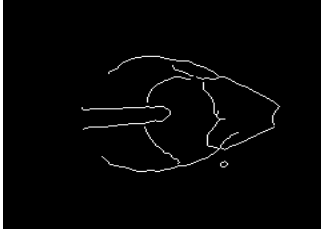
4.2.2 Investigation of Using Conventional Techniques

In general, extracting the weld pool boundary from a single frame falls into the category of image segmentation. Typical techniques for image segmentation are such as edge detection, thresholding, region-based method etc.

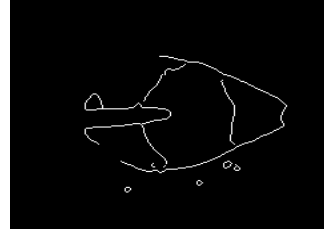
Edge Detection

Fig.4.3 shows typical results of applying Canny edge detector (Canny, 1986). The threshold of Canny edge detector was carefully optimized. The performance of Canny edge detector is rather poor in our application. Only in Fig.4.3(a) the complete weld pool boundary is revealed by edge detection. But unfortunately we meanwhile have to deal with spurious edges which is usually very difficult. In Fig.4.3(b) - (d), Canny edge detector fails to detect the complete boundary.

In fact, by observing the detected edges, we can see that there are different kinds of edges including the weld pool boundary, the arc light boundary and the spurious edges detected in the seam due to the rough surface. Fig.4.4 gives a graphically illustration of different edges detected by Canny edge detector. If only the local intensity change is used to detect the edges, like what Canny edge detector does, all these edges will be extracted. This is undesired. Moreover, as shown in Fig.4.4, we also observe that Canny edge detector failed to detect the yellow part of the weld pool boundary. Therefore it is important to introduce



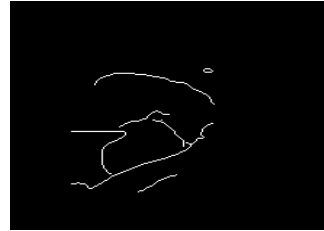
(a) Experiment 1, Frame 53



(b) Experiment 1, Frame 55



(c) Experiment 2, Frame 48



(d) Experiment 2, Frame 81

Figure 4.3: Applying Canny edge detector to the weld pool images. See the text for detailed explanation.

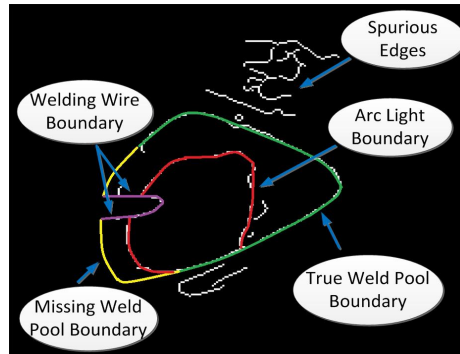


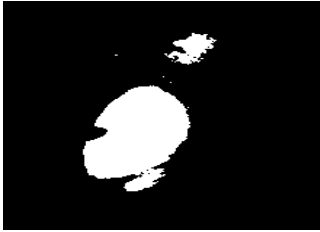
Figure 4.4: An explanation of different kinds of edges in frame 48 of the experiment 2 detected by Canny edge detector. Different edges are marked in different colors. It can be seen that the detected edges contain many spurious ones and noises. In this image, we also observe that one part of the weld pool boundary marked in yellow is missing and could not be detected by Canny edge detector.



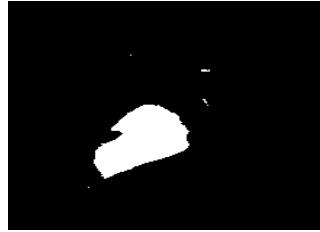
(a) Experiment 1, Frame 53



(b) Experiment 1, Frame 55



(c) Experiment 2, Frame 48



(d) Experiment 2, Frame 81

Figure 4.5: Applying Otsu thresholding to the weld pool images. See the text for detailed explanation.



(a) Experiment 2, Frame 52



(b) Experiment 2, Frame 68

Figure 4.6: Applying thresholding to the weld pool images. The threshold was optimized for our application. See the text for detailed explanation.

prior knowledge into edge detection. This motivates us to employ methods such as deformable models which provide an elegant framework of embedding prior knowledge.

Thresholding

Fig.4.5 shows the results of applying Otsu thresholding techniques (Otsu, 1979). It also fails to identify the weld pool boundary. It should be noted that in Otsu method, the image is assumed to be composed of two regions, foreground and background. An optimum threshold which separates these two classes is computed automatically. This is generally good for developing a fully-automated system. However, its performance is unsatisfactory in our application.

We therefore try regular thresholding with the threshold optimized manually for our application. The results can be found in Fig.4.6. The original images are shown in Fig.4.2(f) and (g). In Fig.4.2(f), the arc light region is pretty small and the performance of the thresholding is rather good. The arc light region in Fig.4.2(g) is otherwise quite large and the thresholding result is unsatisfied. This observation inspired us to the first weld pool boundary tracking algorithm based on inflating balloons. The detail will be discussed in section 5.

4.3 Introduction to Deformable Models

The purpose of this section is to introduce deformable models briefly. We shall explain the basic principles and discuss advantage and disadvantage of different kinds of deformable models. This will serve as the basis of selecting proper deformable models or design a new one if necessary for our application.

Deformable models, also referred to as active contours, snakes in some literatures, have been extensively studied for decades and found numerical applications such as image segmentation, visual tracking (Xu and Prince, 2000). Taking image segmentation as an example, typical methods for segmenting images such as edge detection, thresholding, region-based techniques and so on often suffer from lack of ability of handling prior knowledge. In some circumstances they may be seriously affected by noise and artifacts and even fail completely. Deformable models provide an elegant framework to incorporate prior knowledge.

Generally speaking, deformable models are curves or surfaces that can move under certain *driving forces*. In terms of the representation of the curves, de-

formable models can be classified into two categories: parametric and geometric deformable models. In terms of how to define the *driving forces*, there are different kinds of formulation of deformable models, such as *energy-minimizing formulation*, *dynamic force formulation* and *Bayesian formulation*.

4.3.1 Parametric Deformable Models

One pioneer work of parametric deformable models is due to Kass etc (Kass et al., 1988). In their paper, a snake is defined as an energy-minimizing curve. The characteristic of the curve is given by an energy functional. Assume that \mathcal{C} is a parametric curve which is represented as

$$\mathbf{u}(s) = (x(s), y(s)), s \in [0, 1] \quad (4.1)$$

where s is a parametrization of the curve. The energy functional is then defined as follows.

$$E(\mathbf{u}) = \int_0^1 E_{\text{internal}}(\mathbf{u}) + E_{\text{image}}(\mathbf{u}) + E_{\text{constraint}}(\mathbf{u}) ds \quad (4.2)$$

where E_{internal} denotes the internal energy which stands for regularity or smoothness of the curve. A popular definition is as follows.

$$E_{\text{internal}} = \int_{\mathcal{C}} \alpha \left\| \frac{d\mathbf{u}}{ds} \right\|^2 + \beta \left\| \frac{d^2\mathbf{u}}{ds^2} \right\|^2 \quad (4.3)$$

The first and second order derivatives are related to the tension and stiffness of the curve respectively. α and β are the weights balancing these two terms.

E_{image} denotes the image or potential energy. A typical example for image segmentation is such as

$$E_{\text{image}}(\mathbf{u}) = \frac{1}{\|\nabla(G_\sigma * I)(\mathbf{u})\|^2} \quad (4.4)$$

where G_σ is a Gaussian kernel with standard deviation σ and where $*$ denotes convolution. By minimizing this term, the curve is intended to converge to the edge of the desired objects.

$E_{\text{constraint}}$ denotes the extra constraint energy which offers the user more options to control the curve evolution.

An active contour therefore corresponds to the following functional minimization

problem given certain boundary conditions.

$$\min_{\mathbf{u}(s)} E(\mathbf{u}) \quad (4.5)$$

$$\text{s.t. boundary conditions} \quad (4.6)$$

A necessary condition of minimizing the energy functional is the *Gâteaux* derivative vanishes. Hence, we have the following Euler-Lagrange equation:

$$-\alpha \frac{d^2 \mathbf{u}(s)}{ds^2} + \beta \frac{d^4 \mathbf{u}(s)}{ds^4} + \nabla E_{\text{image}} = 0 \quad (4.7)$$

Deriving an analytical solution to the Eq.4.7 is usually difficult. Instead, it can be solved numerically by treating $\mathbf{u}(s)$ as a dynamic curve, i.e.,

$$\gamma \frac{\partial \mathbf{u}(s, t)}{\partial t} = -\alpha \frac{d^2 \mathbf{u}(s, t)}{ds^2} + \beta \frac{d^4 \mathbf{u}(s, t)}{ds^4} + \nabla E_{\text{image}} \quad (4.8)$$

when $\mathbf{u}(s)$ reaches a steady state, the left hand side of the Eq.4.8 vanishes and we obtain a solution of Eq.4.7. Note that this is equivalent to applying a gradient descend algorithm to Eq.4.2.

It is worth noting that the evolution equation in Eq.4.8 actually shows another formulation of deformable models - *dynamic force* formulation. It allows us to design a deformable model by designing proper forces, instead of energies. It enlarges the set of deformable models that we can have. In fact, some of useful deformable models are derived in terms of dynamic force formulation. For instance in (Xu and Prince, 1998), a novel force field, called *Gradient Vector Flow* (GVF), is derived to enlarge the capture range of deformable models.

There are also some other formulations such as Bayesian formulation. Unlike the dynamic force formulation, Bayesian formulation is equivalent to the energy minimizing formulation. The problem of minimizing an energy functional becomes a *maximum a posterior probability* (MAP) problem. This will facilitate us to benefit from the techniques in the machine learning toolbox.

Parametric deformable models have been widely used in many applications. The advantage is that they are often straightforward to implement and require much less computation time compared to geometric deformable models. We will explain more after introducing geometric deformable models. One big constraint to these methods is that they can not deal with topological changes, i.e., splitting or merging without special mechanisms. Another limitation is that parametric

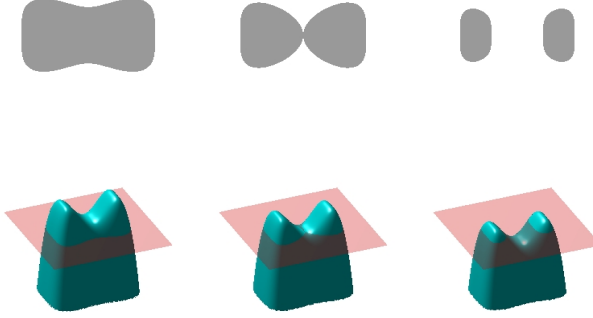


Figure 4.7: An illustration of level set functions (Alexandrov, 2004). Instead of a 1D curve in parametric deformable models, a 2D surface here is maintained and manipulated in a geometrical deformable model. The topological change of a 1D curve can be easily achieved by manipulating the corresponding 2D surface. This is a great advantage of geometrical deformable models.

deformable models often need reparametrization of the curve frequently during evolving. If they are not properly reparameterized, numerical instability occurs and may cause the failure of the whole evolving. As we have been making use of parametric deformable models in our application, we will discuss in details how to stabilize the evolution in Chapter 5.

4.3.2 Geometric Deformable Models

Geometrical deformable models were first proposed independently by Caselles et al. (Caselles et al., 1993) and Malladi et al. (Malladi et al., 1995; Osher and Fedkiw, 2002). The essential idea is that the curve \mathcal{C} is represented implicitly as a zero level set of a higher-dimensional function, a level set function, ϕ , see Fig.4.7.

$$\phi(t, \mathbf{u}(t)) = 0 \quad (4.9)$$

where t is an artificial evolving time step. Differentiating both sides over t of the Eq.4.9 leads to

$$\frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial t} = 0 \quad (4.10)$$

Assume that the curve moves in the flow field given as follows

$$\frac{\partial \mathbf{u}}{\partial t} = FN \quad (4.11)$$

where F is a vector field. $N = -\frac{\nabla \phi}{\|\nabla \phi\|}$ is the inward unit normal to the curve. The corresponding evolving equation of ϕ can be deduced as follows:

$$\frac{\partial \phi}{\partial t} = -\frac{\partial \phi}{\partial \mathbf{u}} \cdot \frac{\partial \mathbf{u}}{\partial t} \quad (4.12)$$

$$= -\nabla \phi \cdot FN \quad (4.13)$$

$$= F\|\nabla \phi\| \quad (4.14)$$

This relationship between evolving a parametric curve and evolving the corresponding level set function shows that one can design a geometric deformable model, or level set method, by defining force fields which drives the curve. We can also define an energy functional $E(\phi)$ for the level set function ϕ directly. The partial differential equation(PDE) that governs the evolution of ϕ can be derived by calculus of variations:

$$\frac{\partial \phi}{\partial t} = -\frac{\partial E(\phi)}{\partial \phi} \quad (4.15)$$

Level set methods have the advantage of handling the topological change automatically. The downside is that level set methods are computationally expensive since in each iteration all the grid points representing ϕ need to be updated. Even for the *narrow-band* scheme (Adalsteinsson and Sethian, 1995), the grid points is much more than the control points used in a parametric deformable model. It is worth noting that A parallel version of a geometrical deformable model using Graphics Processing Unit(GPU) may greatly reduce the processing time (Lefohn et al., 2005; Roberts et al., 2010).

4.4 Weld Pool Tracking Using Deformable Models

In the previous sections, we show that weld pool boundary extraction using conventional image analysis techniques is rather difficult. Low-level techniques yield unsatisfactory results partially due to the lack of the ability of incorporating prior knowledge. This motivates us to investigate the possibility of using deformable models. In this section, we shall discuss two critical questions on applying deformable models to weld pool boundary extraction.

1. Which is the proper approach in our application, parametric or geometric deformable models?
2. How shall we design proper energy functionals or driving forces for our particular application?

Based on the discussion in the previous section, the answer to the first question is rather straightforward. Considering the fact that topological change does not exist in weld pool boundary tracking and the requirement on computing resources, parametric deformable models are more proper in our application.

As the answers to the second question, we will present three different approaches based on deformable models for weld pool boundary tracking in the next few sections.

4.4.1 Evaluation Criterion

Since the final goal of extracting weld pools is to control the welding processes or for online quality inspection, the ultimate criteria is to make use of these visual information and examine the final welding quality. However, it is still important for us to be able to evaluate the proposed weld pool tracking algorithms from purely image analysis point of view.

As the weld pool shape may change a lot from frame to frame and in different experiments, a uniform ground truth does not exist. The evaluation of the *accuracy* of the proposed algorithms therefore mostly relies on the welding expert inspection. On the other hand, to quantitatively measure the performance of a weld pool boundary tracking algorithm compared to welding experts inspection, we have provided a reference boundary for each frame selected and presented in this dissertation by manually segmenting the weld pool boundary. This serves as a quantitative measurement of the *accuracy* of an algorithm, besides welding expert inspection. The detailed discussion of representing and comparing weld pool boundaries as well as manually segmentation of weld pool boundaries are presented in Appendix B.

Note that even though we still refer to a reference boundary as a *ground truth*, it is only a quantitative representation of human inspection, not the *true* weld pool boundary. The boundary extracted by algorithms can be more accurate than the one extracted by human experts especially when the boundary is visually weak.

In this dissertation, we select three typical videos which were captured in different experiments and contain very different weld pools and examine the performance of the proposed algorithms in terms of both *accuracy* and *speed*.

1. Experiment 1. The first experiment was made as a root path short-circuit GMAW on a narrow v-groove seam. The torch approximately followed the center line of the seam without oscillation. The characteristics of the weld pool images are: First, the weld pool is almost symmetry and slowly changed from frame to frame. Second, strong arc light exists above the weld pool region.
2. Experiment 2. The second experiment was again made as a root path short-circuit GMAW on a narrow v-groove seam, but in this case the path of the torch was a little away from the center line. The consequence is that the arc is generated and the weld pool is deposited on one side of the seam. On the images, the weld pool is asymmetry and slowly changed.
3. Experiment 3. The third experiment was made as a root path short-circuit GMAW on a wide v-groove seam. The torch was oscillated. In this case, the arc was generated by two sides of the seam alternatively. Both the position and the shape of the weld pools change dramatically from frame to frame. It should be noted that it is possible to reduce the visual change of the weld pool shape between two successive frames by increasing the speed, i.e., frame per second, of the camera. However, the cost will increase too. It is therefore preferable that the image analysis algorithms have the capacity of tracking weld pool boundaries varying a lot from frame to frame.

4.5 Approach I: Inflating Balloons

4.5.1 Method

The first attempt we made to extract the weld pool boundary is based on the most basic deformable models - active contours proposed by Kass etc (Kass et al., 1988). A brief introduction to this model has been given in section 3. Now we shall select or design proper energy functionals particularly for our application.

4.5.1.1 Design of Energy Functionals

Deformable models can only perform well when proper energy functionals or dynamic forces are selected or designed. In the case of snakes proposed by Kass etc, there are two basic energy terms needed to be specified: the internal force and the image force. First, we adopt the most popular definition of the internal force which is a weighted combination of the first order derivative and the second order derivative, as shown in Eq.4.3. Second, to track the weld pool boundary revealed by image intensity edges we use the following image force

$$E_{\text{image}} = -\|\nabla(G_\sigma * I)(\mathbf{u}(s))\|^2 \quad (4.16)$$

The capture range of this image force is rather small. In other words, the initial curve has to be put close to the weld pool boundary in order to make sure the image force takes effect. This is one reason that manually initialization is required in many image segmentation applications. However, this is unacceptable in our application. One possibility of enlarging the capture range of a snake model is to benefit from the *shrinking* effect of the internal force. It is important to note that the primary effect of the internal force is to smooth the curve, bridge the gap and so on. However, it has a side effect: it will shrink the curve. Without any other forces, the curve will converge towards to a point. The cost of taking advantage of this side effect is that the risk of losing the details of the image boundary will increase.

Enlarging the capture range of deformable models is of great importance for applications that can not allow human intervention. A large capture range implies the insensitivity or robust to the initialization. Many techniques for this purpose have been proposed. For instance, Cohen (Cohen, 1991) proposed to add an extra force called *inflating force* so that the resulting deformable model behaviors like an inflating balloon. Adding this inflating force is equivalent to maximizing the area Ω enclosed by the contour.

$$E_{\text{inflating}} = \int \int d\Omega \quad (4.17)$$

Other techniques that can increase the capture range of the deformable models exist, such as gradient vector flow (Xu and Prince, 1998) or region-based techniques (Chan and Vese, 2001). Computing the GVF is time-consuming and it does not meet the real-time requirement. Therefore, it is not proper for our application. On the region-based techniques, we will discuss more later.

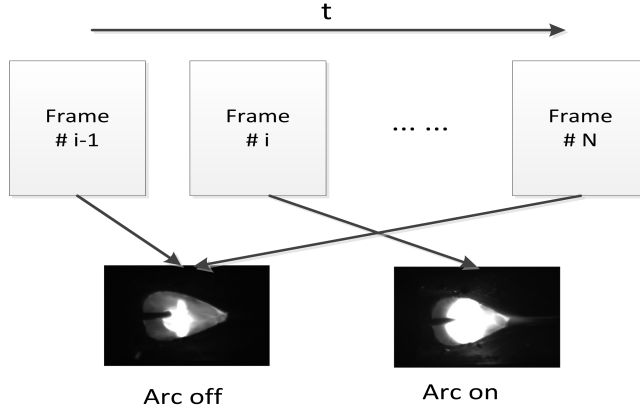


Figure 4.8: The frames are categorized into two groups: arc-on frames and arc-off frames. This takes advantage of the fact that the arc is generated or disappeared periodically in short-circuit arc welding.

4.5.1.2 Periodically Initialization Strategy

In general the weld pool boundary changes slowly making it possible to use the final curve position from the previous frame as the initial position for the preceding frame. Often the image dependent force will be small at positions away from the true boundary. In particular, the gradient magnitude will be large at the arc light boundary. Especially when the arc is generated, the interference is very intensive making the arc light cover the true boundary of the weld pool and causing an instability of the tracking algorithms. Also, reflections may cause false edges to which the snake may falsely be attracted.

We therefore propose a periodically bootstrap strategy for the initialization of the active contours based on a binary classification of the frames as arc-on or arc-off. In Fig.4.8, it is shown that arc-off frames and arc-on frames are categorized according to their intensities. Frames with an average intensity below a threshold will be regarded as arc-off frames. Otherwise, arc-on frames. This takes advantage of the fact that the arc is generated or disappears periodically in short-circuit arc welding.

Fig.4.9 shows the average intensity for a sequence of frames. The arc-off frames marked by circles are solitude and spread more or less evenly across the sequence. This makes them useful for initialization frames. More important however, is that the true weld boundary is much more easily detectable in arc-off-frames.

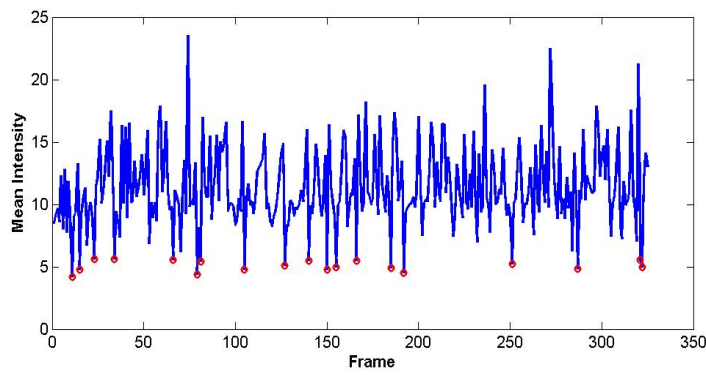


Figure 4.9: This is a graphically illustration of categorizing frames into arc-off frames and arc-on frames. The arc-off frames are marked by circles. The curve shows the average intensity of all frames.

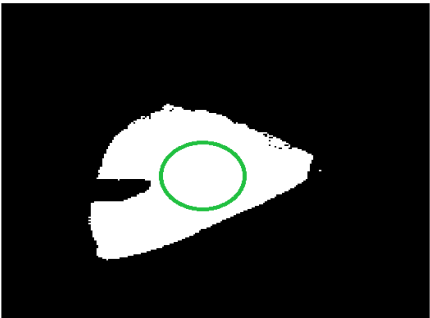


Figure 4.10: This figure illustrates the initialization of an inflating balloon in an arc-off frame. This is a binary image by thresholding the original frame. The green circle inside the white region represents the initial position of the inflating balloons.

In the case of arc-off frames, the arc light region is relatively small and will not cover the boundary of the weld pool. In this case, we threshold the image and use an inflating balloon to extract the weld pool boundary. The initial curve is automatically positioned inside the weld pool region, as shown in Fig.4.10. This is done by computing the image moment. Starting from this initial circle, the inflating balloon is inflated and converges to the boundary of the weld pool.

For arc-on frames, more intensive arc light can be observed, occasionally covering the weld pool boundary. We use an active contour without the inflating force and the tangent force to extract the weld pool boundary. In this case, we rely on the contour position from the previous frame as a good initial position. The pseudo-code of the proposed method can be found in Algorithm 3.

```

input : A Frame  $I_i$ 
output: Weld Pool Boundary  $\mathcal{C}_i$ 
Check whether the arc is off in the current frame.;
if true then
    // If the arc is off, the inflating balloon is employed.
    Threshold the frame  $I_i$  into a binary image  $I_i^b$ ;
    Compute and position the initial curve by computing the image moment. ;
    Run Inflating Balloons on the binary image  $I_i^b$ .;
else
    // The arc is on, the regular active contour is employed.
    The initial curve is selected to be the final curve in the previous frame.;
    Run Active Contours without an inflating force. The initial curve is the
    final curve that ;
end

```

Algorithm 3: Weld Pool Boundary Tracking based on Inflating Balloons

4.5.1.3 Advantage and Disadvantage

The above weld pool tracking algorithm based on inflating balloons is fast, easy to implement and requires no off-line training. On the other hand, it also has a few constraints. First, it can only work in short-circuit arc welding, since we take the advantage of the fact that the arc is on and off periodically. Second, it relies on the performance of thresholding. If thresholding fails, the inflating balloon will not yield satisfactory results. Third, for arc-on frames, we use the strategy of using the result in the previous frame to initialize the current active contour. This strategy may fail if the weld pool boundary varies dramatically from frame to frame. Examples and more discussions will be shown in the next subsection.

Table 4.1: Parameters used in Approach I

Parameters	Symbol	Value
Internal force weight : Tension	α	0.01
Internal force weight : Stiffness	β	0.01
Spacial step size	Δs	1
Temporal step size	Δt	0.01
The inflating force weight	k	0.002
Threshold for arc-on/off frame categorization	λ_1	6
Threshold for the arc-off frames	λ_2	20
Number of control points	N	100
Iterations of Inflating Balloons	$N_{\text{arc-off}}$	3000
Iterations of Kass's Snakes	$N_{\text{arc-on}}$	200

4.5.2 Experiments

In this section, we shall present the experiments and results of applying the above weld pool boundary tracking algorithm based on inflating balloons. The parameters used in the proposed methods are shown in table 4.1. We have tested the approach 1 on three different experiments. A number of frames were segmented manually as discussed in Section 4.4.1 and Appendix B. Compared to these *ground truths*, statistical results are generated and presented.

4.5.2.1 Experiment 1

Fig.4.12 presents the statistical results of testing this approach in Experiment 1. Fig.4.12(a) shows box plots of the error in each frame. It can be seen that the average error in each frame is around 3 pixels, despite the outliers. Fig.4.12(b) shows the error at each control point. Both mean and deviation are rather small, except at the control points 17 ~ 20. It should be pointed out that the outliers in fact correspond to the control points 17 ~ 20 which are located at the weld wire boundary (see Fig.4.4). When generating the ground truth, the weld wire boundary is excluded deliberately for the resolution reason. On the other hand, the weld pool boundary tracking approaches do extract the wire boundary. As a consequence, big errors appears which is however not a sign of bad performance. In fact, big errors here indicate the weld wire boundary has been captured.

Furthermore, we showed image analysis results of eight frames in Fig.4.17.

Among them, Fig.4.17(a) and (e) are labeled as arc-off frames. All the others are labeled as arc-on frames. For all the frames, the weld pool boundaries are successfully extracted. It can also be observed that the weld pool shape does not change too much from frame to frame. This does meet the assumptions of weld pools changing slowly and the algorithm performs well.

4.5.2.2 Experiment 2

In Experiment 2, the statistical results are shown in Fig.4.13. It can be seen that the performances are rather good in most frames, especially those frames which contain less arc light, such as frame 1, 5, 8. The performances are less satisfactory in the frames where strong reflection of the arc light occurs such as frame 3, 13, 15. Fig.4.18 shows image analysis results and serves as a complement to the statistical results.

Among all eight frames we show in Fig.4.18, (a) and (c) were categorized as arc-off frames. On these two frames, the algorithm captured the weld pool boundaries very accurately. For the arc-on frames (d) (e) and (f), the performances of the algorithm are relatively good. However, we can also observe unacceptable results in (g) and (h). Fig.4.15 shows the details where the algorithm failed. The truth boundary of the weld pool is marked in yellow. The extracted boundary by the proposed algorithm is marked in green. The algorithm was trapped by the strong spurious edges and failed.

4.5.2.3 Experiment 3

As discussed previously, torch oscillation is required in experiment 3. The weld pool shape changes more rapidly. The statistical results and image analysis results are presented in Fig.4.14 and Fig.4.19, respectively.

The statistical results show that the approach performs very well in Experiment 3. The errors in almost of all frames are within 5 pixels with an exception of frame 106 shown Fig.4.19(d). The failure on frame 106 is mainly due to the image quality here is rather poor. We show a zoomed in picture of frame 106 in Fig.4.16. In the red box, the weld pool boundary is in fact indistinct and not very-well defined by intensity changes or textures. It is therefore difficult to extract the weld pool boundary only from this frame. One possible solution is to incorporate shape prior into the framework of active contours. However, from the control point of view, the control commands are normally not generated by processing only one frame especially when the image quality varies from frame

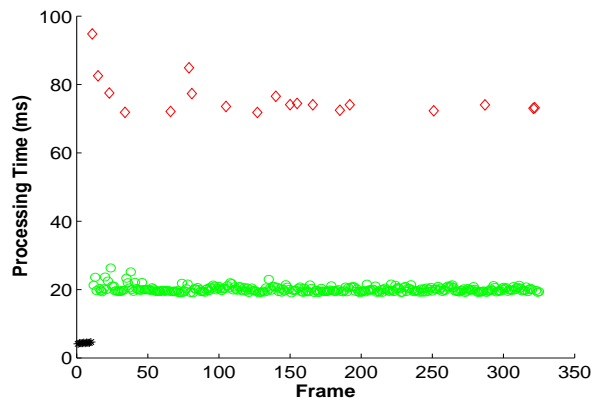
to frame. It is not necessary to make use of all the frames. We can simply detect and discard these frames.

4.5.2.4 Speed

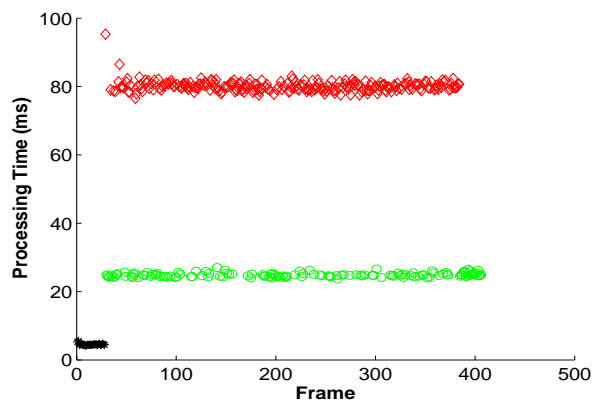
This algorithm is implemented in C++ and fast enough to be run in real time on an ordinary desktop PC. In Fig.4.11, the speed of the proposed algorithm tested on experiment 1 and 3 is shown. Approximately, it takes 80ms (12.5Hz) to process the arc-off frames and 20ms (50Hz) to process the arc-on frames. It can also be seen that only a small amount of frames are classified as arc-off frames in experiment 1. This is consistent with our observation. In experiment 3, on the other hand, the amount of arc-off frames is significantly more than experiment 1 due to the fact that the arc light is relatively small. This will increase the processing time.

4.5.2.5 Concluding Remarks

In summary, the tracking algorithm based on inflating balloons is simple, easy to implement and needs no off-line training. The test on three different experiments confirms that it can handle different types of weld pools. The drawback of this algorithm is that it relies on thresholding on arc-off frame to obtain the initialization for arc-on frames. In other words, this tracking algorithm has no ability to process the arc-on frames directly. The limitation is that it can only be used in short-circuit arc welding. In the case of spray arc welding or if extremely strong arc exists, it may fail. Therefore, it is highly desirable to design weld pool tracking algorithms that can process the arc-on frames directly.

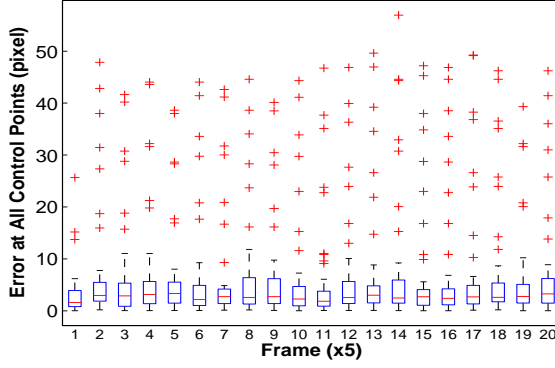


(a) Experiment 1

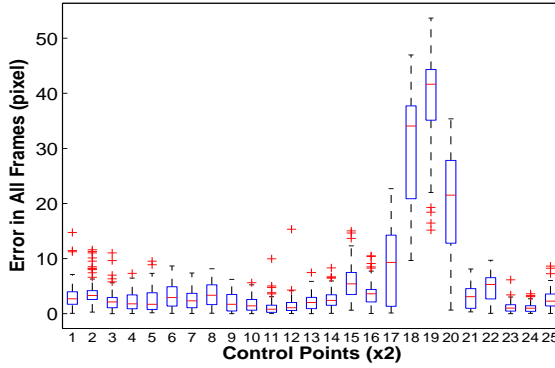


(b) Experiment 3

Figure 4.11: The figure shows the time of processing arc-off and arc-on frames. The arc-off frames are marked in red diamonds. The arc-on frames are marked in green circles. The frames before the arc is generated are marked in black stars. The time of processing arc-off frames is approximately 80ms. For arc-on frames, it is 20ms.

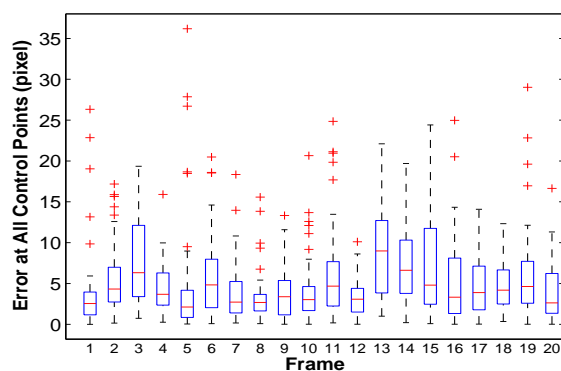


(a) Error at each frame. Note that the box plots were made every five frames.

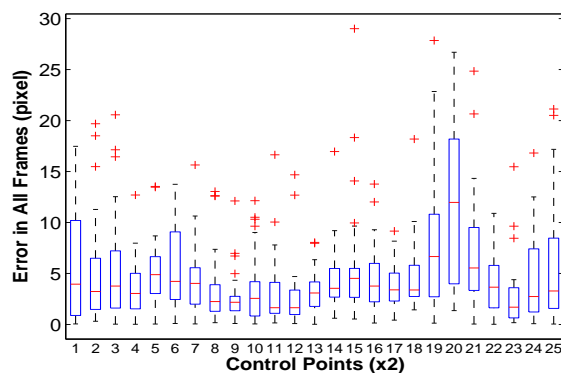


(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.12: Statistical results of testing Approach I in Experiment 1. The box plots are computed from 100 frames. See text for explanation. The definition of *error* can be found in Appendix B. It is worth noting that in the ground truth, the weld wire boundary is excluded deliberately. On the other hand, the algorithm does extract the wire boundary along with weld pool boundary. This gives rise to the large errors in the control points 17 ~ 20 in (b) which also corresponds to the outliers shown in (a). This applies to all the subsequent figures in this chapter when presenting statistical results of weld pool boundary tracking.

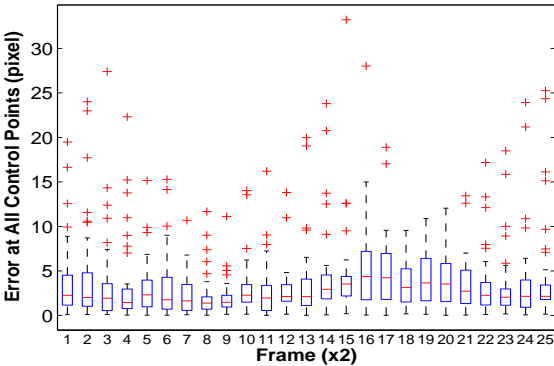


(a) Error at each frame. Note that the box plots were made every two frames

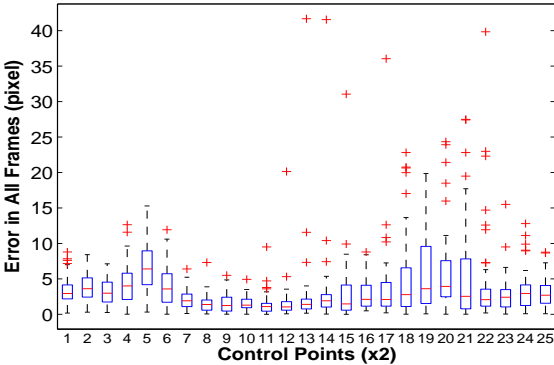


(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.13: Statistical results of testing Approach I in Experiment 2. The box plots are computed from 20 frames. See text for explanation.



(a) Error at each frame.



(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.14: Statistical results of testing Approach I in Experiment 3. The box plots are computed from 50 frames. See text for explanation.

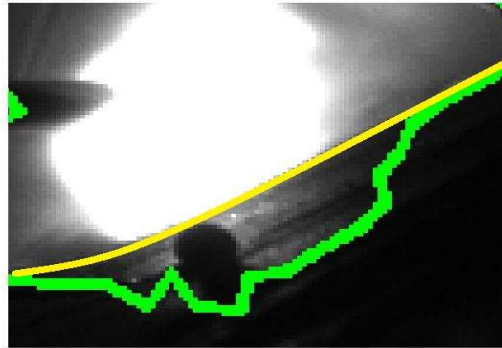


Figure 4.15: A zoomed in figure of Fig.4.18(h). The truth boundary of the weld pool is marked in yellow. The extracted boundary by the proposed algorithm is marked in green. The algorithm failed in this frame because there are more than one strong edges. Unfortunately, the algorithm did capture an edge, yet false weld pool boundary.

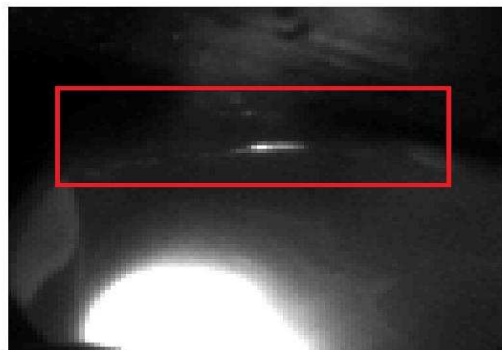


Figure 4.16: A zoomed in picture of Fig.4.19(d) where the tracking algorithm based on inflating balloons failed. It can be observed that the boundary of the weld pool in the red box is indistinct.

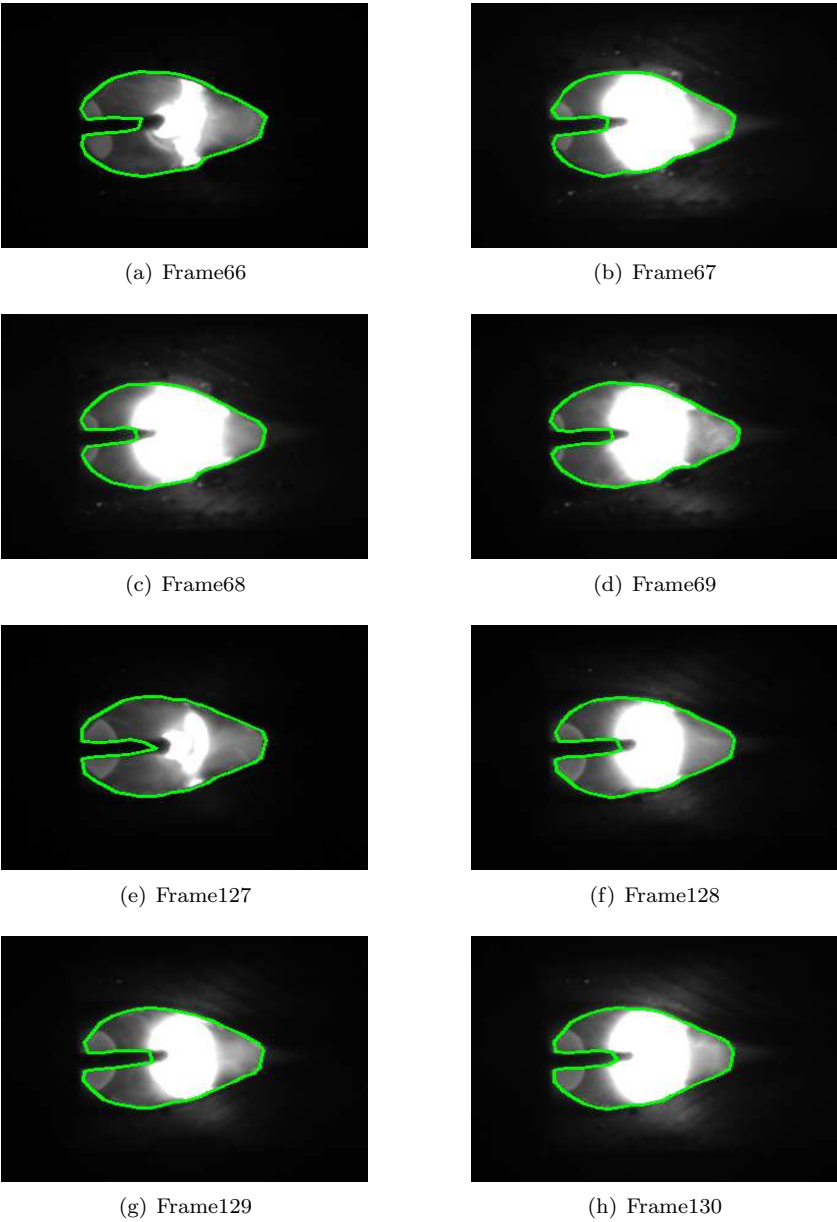


Figure 4.17: Experiment 1. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves.

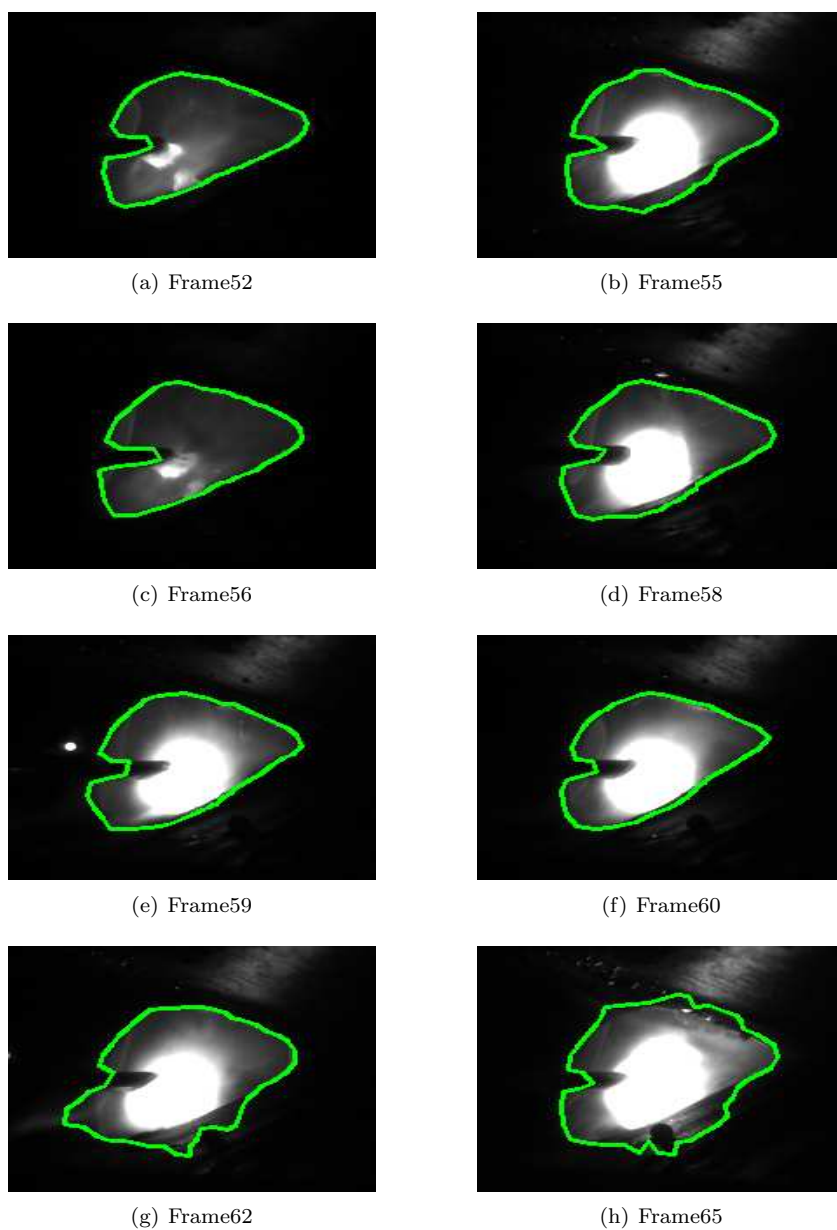
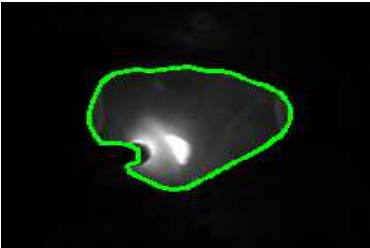
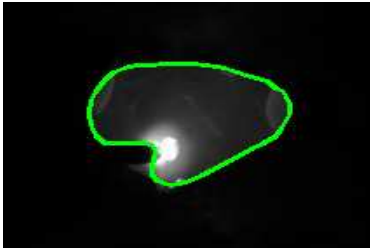


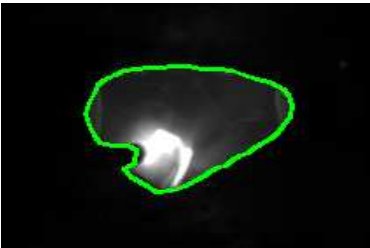
Figure 4.18: Experiment 2. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves.



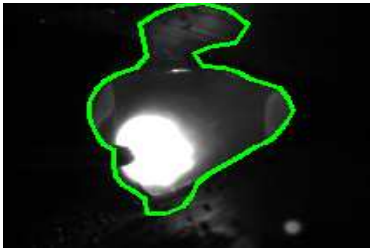
(a) Frame103



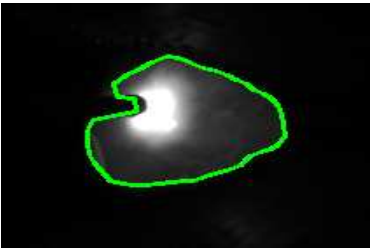
(b) Frame104



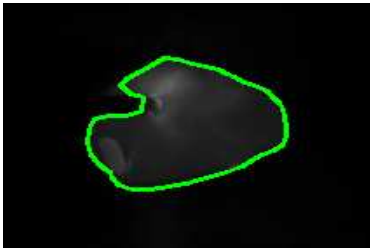
(c) Frame105



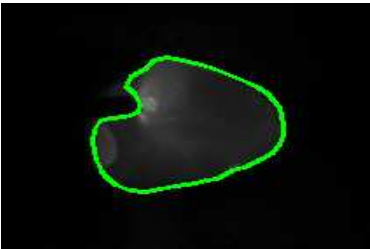
(d) Frame106



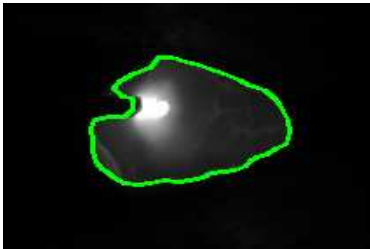
(e) Frame131



(f) Frame132



(g) Frame133



(h) Frame134

Figure 4.19: Experiment 3. Torch oscillation is required. The extracted weld pool boundary are marked in green curves.

4.6 Approach II: Active Contours Driven by Region based Statistical Models

4.6.1 Method

In last section, we discuss an approach that uses inflating balloons and active contours proposed by Kass (Kass et al., 1988). We also propose a special periodical initialization technique to initialize these models. We have shown that the algorithm presented in approach 1 is not able to process the arc-on frames directly. In addition, in some cases it may be attracted by the false boundary which has a very strong edge response and therefore performs unsatisfactorily. The weakness is mainly due to the fact the image force as in Eq.4.16 is too general and not particularly designed for our application. Moreover, the capture range of this image force is too small. To tackle this problem, we propose to use an image force that is based on region based statistical models. These models are specifically trained for our application. This new potential force has a large capture range which allows us to position the initial contour almost anywhere in the image.

Active contours in many applications are superior to conventional low-level image segmentation techniques due to the ability of handling prior knowledge elegantly. It is important to point out that *prior knowledge* can be interpreted rather broadly ranging from such as spatial and temporal smoothing to so-called *intensity prior* and *shape prior*.

For instance, the prior knowledge included in the active contours used in approach I is only the spatial smoothing. With this spatial smoothing, active contours have the ability to fill in the gap and tolerant a small amount of noise naturally, and finally provide a smooth and continuous curve. This ability comes from the minimization of the internal force which serves the purpose of smoothing the curve spatially.

Beside spatial smoothing, we do have other prior knowledge, or observations, related to our specific applications such as

1. The intensity of arc light region in the image is very close to purely white.
2. Typically we have

$$\text{Intensity}(\text{arc light}) > \text{Intensity}(\text{weld pool}) > \text{Intensity}(\text{background})$$

3. A weld pool should roughly look like an ellipse.

The first and second items are about the distribution of the intensity of different regions, so called intensity prior. These are important and significant information in our application. We shall discuss how to model intensity prior systematically and how to embed this prior into the framework of active contours.

The third observation is in fact related to *shape prior*. A possible solution of modeling this prior is to learn a statistical shape model from pre-labeled weld pool images. Currently It is not included in current works of ours and we leave the investigation of using shape prior to the future work.

4.6.1.1 Active Contours Driven by Region-based Statistical Models

The concepts of region-based forces arising from the statistical modeling of different regions were first proposed by Chan and Vese in (Chan and Vese, 2001). In their works, the following energy functional was defined:

$$E(c_1, c_2, \mathcal{C}) = \mu \cdot \text{length}(\mathcal{C}) + \nu \cdot \text{area}(\Omega_{\text{in}}) + \lambda_1 \int_{\Omega_{\text{in}}} |I(\mathbf{x}) - c_1|^2 d\mathbf{x} + \lambda_2 \int_{\Omega_{\text{out}}} |I(\mathbf{x}) - c_2|^2 d\mathbf{x} \quad (4.18)$$

The first two terms are corresponding to internal forces. The last two terms are so-called region-based forces. Taking partial derivatives with respect to c_1 and c_2 and making them vanish, we can prove that c_1 and c_2 are equal to the average intensity of two regions. In what follows, we will refer to this region-based force as CV model.

The CV model has an advantage that it is robust to initialization and noise. By simple calculation, we can see that the force can be big even far away from the boundary. In contrast, the image force as shown in Eq.4.16 may be very small when it is away from the boundary and is not able to drive active contours.

The limitation of the CV model is that it implicitly assumes that the intensities in two regions are Gaussian distributed. If the images do not meet this assumption, the performance may be poor. Therefore, many researchers attempt to generalize the original CV model (Paragios and Deriche, 2002b; Olivier et al., 2008b; Chen and Radke, 2009). Space does not allow a full review. However for the present approach the most relevant improvement is due to (Olivier et al., 2008b). In (Olivier et al., 2008b), the authors proposed an active contour driven explicitly by supervised binary classifiers, more specifically a k-nearest neighborhood, a support vector machine and a neural network. The idea is that binary classifiers are trained to distinguish different regions and then drive the active contour toward or away from the desired boundary.

Another important work is due to Zhu etc (Zhu and Yuille, 1996). In their work in 1996, the authors explicitly proposed that image segmentation can be done by region competition which is essentially a classifier built upon statistical models.

It should be pointed out that almost all region-based forces are defined originally for level set methods where regions can be naturally handled. For parametric active contours as we use in our applications, handling the regions is not as convenient as geometric active contours and it can be time-consuming. One elegant solution is to apply Green's Theorem and transform the region-based energy integral to line integral.

Now we shall derive the general form of active contours that driven by statistical models. Suppose that we are dealing with binary segmentation problem. The entire image is separated into two regions, denoted as Ω_{in} and Ω_{out} , and the boundary \mathcal{C} . In each region, we further assume that the intensity, considered as a random variable, at each pixel (x, y) are independently and identically distributed, denoted as p_{in} and p_{out} , respectively. The likelihood of the whole image given the segmentation \mathcal{C} is computed by

$$\begin{aligned} P(I | \mathcal{C}) &= \prod_{\mathbf{x} \in \Omega} p(I(\mathbf{x}) | \mathcal{C}) \\ &= \prod_{\mathbf{x} \in \Omega_{in}} p_{in}(I(\mathbf{x})) * \prod_{\mathbf{x} \in \Omega_{out}} p_{out}(I(\mathbf{x})) \end{aligned} \quad (4.19)$$

To maximize this likelihood is equivalent to minimize its negative logarithm. Then we have :

$$\begin{aligned} E &= - \int_{\Omega_{in}} \log p_{in}(I(\mathbf{x})) d\mathbf{x} - \int_{\Omega_{out}} \log p_{out}(I(\mathbf{x})) d\mathbf{x} \\ &= - \int_{\Omega_{in}} \log p_{in}(I(\mathbf{x})) d\mathbf{x} - \int_{\Omega} \log p_{out}(I(\mathbf{x})) d\mathbf{x} + \int_{\Omega_{in}} \log p_{out}(I(\mathbf{x})) d\mathbf{x} \\ &= - \int_{\Omega_{in}} \log p_{in}(I(\mathbf{x})) d\mathbf{x} + \int_{\Omega_{in}} \log p_{out}(I(\mathbf{x})) d\mathbf{x} - \text{constant} \\ &= - \int_{\Omega_{in}} \log \frac{p_{in}(I(\mathbf{x}))}{p_{out}(I(\mathbf{x}))} d\mathbf{x} - \text{constant} \end{aligned} \quad (4.20)$$

This term can be an energy functional term for the inclusion of the intensity prior, denoted as $E_{\text{intensity prior}}$.

To minimize this energy functional, we first use Green's Theorem to transform the region integral to line integral, and then apply calculus of variation. Details can be found in the appendix or (Zhu and Yuille, 1996). The derived gradient

flow is as follows:

$$\log \frac{p_{\text{in}}(I(\mathbf{x}))}{p_{\text{out}}(I(\mathbf{x}))} \parallel \frac{d\mathbf{x}}{ds} \parallel \vec{\mathbf{n}} \quad (4.21)$$

where $\vec{\mathbf{n}}$ denotes the unit vector which is normal to the curve and points outward. The corresponding evolution equation of the curve \mathcal{C} is

$$\frac{\partial \mathbf{x}}{\partial t} = -\log \frac{p_{\text{in}}(I(\mathbf{x}))}{p_{\text{out}}(I(\mathbf{x}))} \parallel \frac{d\mathbf{x}}{ds} \parallel \vec{\mathbf{n}} \quad (4.22)$$

So the next problem is how to model and estimate the intensity distributions over different regions $p_{\text{in}}(I(\mathbf{x}))$ and $p_{\text{out}}(I(\mathbf{x}))$.

Remark 1 . The Eq.4.22 can be further generalized as follows:

$$\frac{\partial \mathbf{x}}{\partial t} = -\lambda \cdot \varphi(\mathbf{x}) \cdot \parallel \frac{d\mathbf{x}}{ds} \parallel \vec{\mathbf{n}} \quad (4.23)$$

where λ is a weighed factor. $\varphi(\mathbf{x})$ is defined as follows:

$$\varphi(\mathbf{x}) = \begin{cases} -1 & \text{if } \mathbf{x} \in \Omega_{\text{in}}, \\ 1 & \text{if } \mathbf{x} \in \Omega_{\text{out}}, \\ 0 & \text{if } \mathbf{x} \in \mathcal{C}. \end{cases} \quad (4.24)$$

Note that $\varphi(\cdot)$ can actually be considered as a classifier. The Eq.4.22 is a variant which uses a classifier based upon maximum likelihood.

Remark 2 . In Eq.4.22, the statistical models are only built upon the intensity. In fact, it is possible to use more features such as gradient, texture features to represent each pixel. It will of course cost us more to process each pixel. In this section, only the intensity of each pixel is used as the representation, hence called *intensity prior*.

4.6.1.2 Statistically Modeling of the Weld Pool Images

Following the assumption we made before, the whole weld pool image can be separated into two regions: Ω_{in} and Ω_{out} . The aim of this section is to select proper means to model the intensity distributions of these two regions. The distributions may be as simple as Gaussian as in the CV model. More complex models such as Mixture of Gaussian or kernel density estimation techniques are also possible choices.

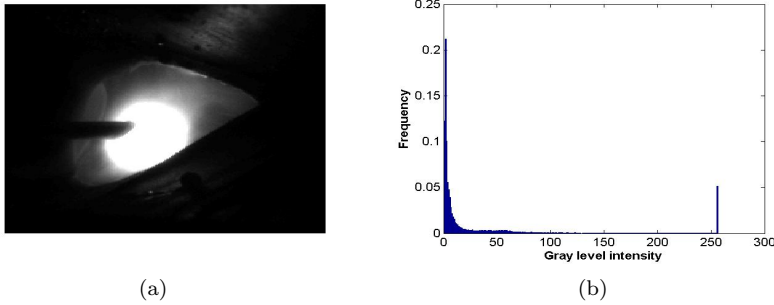


Figure 4.20: (a)The original image. (b)The histogram. The histogram shows that the intensities in the arc light region are very high, and the intensities in the weld pool region are small.

Histogram

Fig.4.20 presents the histogram of the whole weld pool image. To further investigate the histograms of different regions, we manually segment the weld pool images into the foreground and the background and compute the histograms respectively, as shown in Fig.4.21.

The Euclidean distances between two histograms are computed and presented as a distance matrix, see Fig.4.22. The first 17 rows or columns are foreground histograms, the rest are background histograms. The similarity within one group and the dissimilarity between two groups can be easily observed in this distance matrix. This observation shows that the overlap between the histograms of the foregrounds and the backgrounds is small. Now we shall discuss how to model different regions using statistical models.

Mixture of Gaussian

In our applications, we made an attempt to use the Chan-Vese model (Chan and Vese, 2001) for weld pool boundary tracking. The results can be found in Fig.4.23. The CV model performs badly. This can be expected since in this application the intensity distribution of the foreground which is composed of arc light and weld pool is not Gaussian. The results show that Gaussian distribution is insufficient to model the regions in our application. Since in our case there are actually two different light sources, the arc and the weld pool, it is natural to use a mixture of Gaussian to model the foreground and background regions.

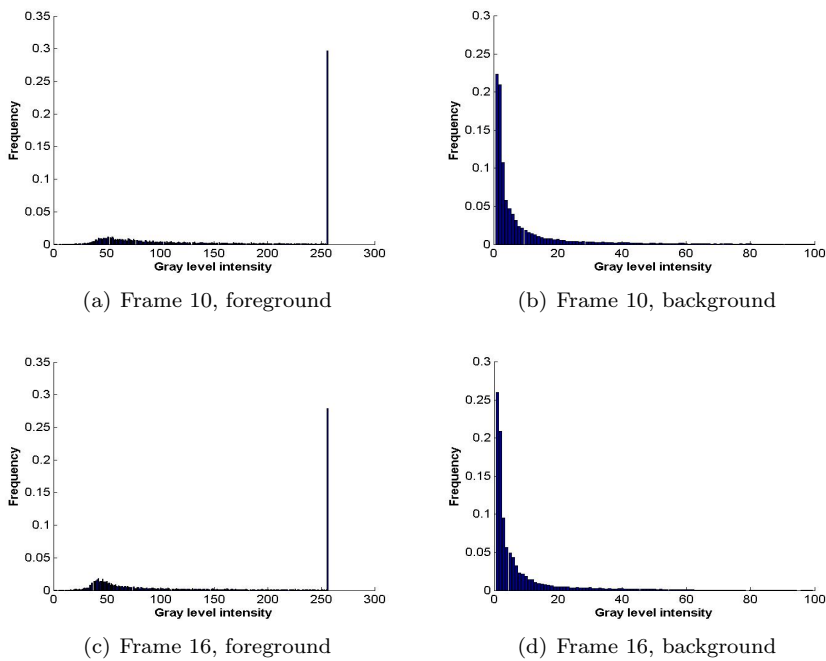


Figure 4.21: The histograms of the background and the foreground in two frames.

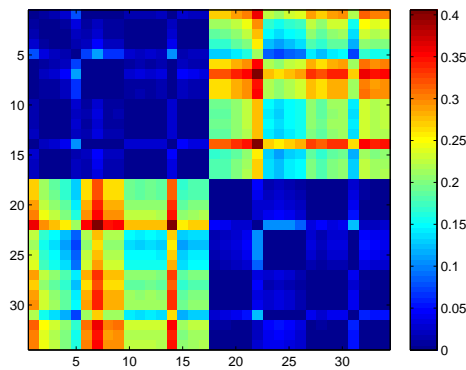


Figure 4.22: The Euclidean distances between each two histograms are computed and presented as a distance matrix. The first 17 rows or columns are foreground histograms, the rest are background histograms. The similarity within one group and the dissimilarity between the groups can be easily observed in this distance matrix.

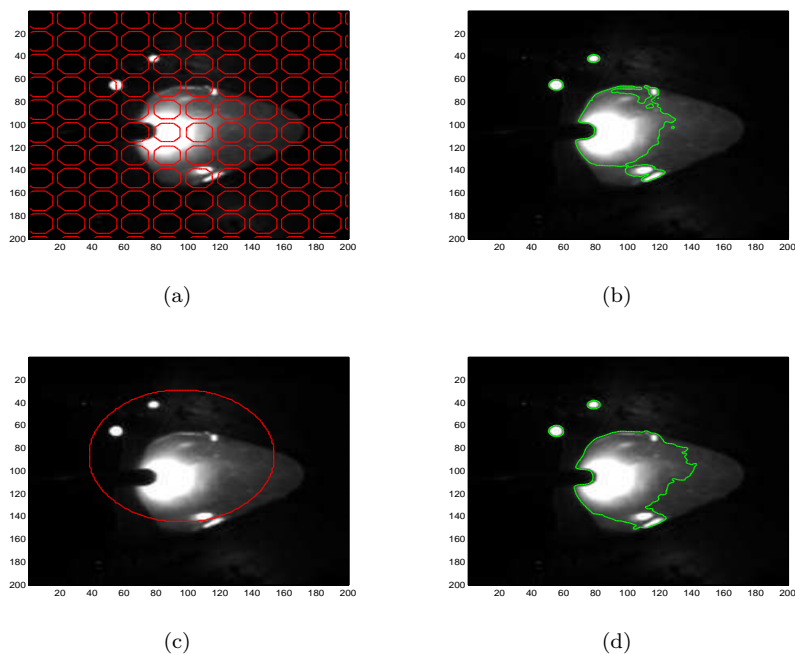


Figure 4.23: Weld pool boundary extraction using the Chan-Vese model. Iterations 1000. (a) Initialization with a large number of spoiled regions. (b) Final segmentation. (c) Initialization with a big circle. (d) Final segmentation. The results are unsatisfactory, see text for more explanation.

We manually segmented the weld pool region and the surrounding and learned the mixture of gaussian model respectively.

The Gaussian mixture models are commonly-used parametric modeling tools for density estimation (Bishop, 2007). The Gaussian mixtures arising from the effort to enrich the classes of density models are constructed as a linear superposition of Gaussian components in the form

$$p(\mathbf{x}) = \sum_{i=1}^N \pi_i \mathcal{N}(\mathbf{x} | \mu_i, \Sigma_i) \quad (4.25)$$

Here N is the number of the Gaussian components. $\pi_i, i = 1, \dots, N$ are the mixing coefficients which satisfies that

$$\begin{aligned} 0 \leq \pi_i &\leq 1 \\ \sum_{i=1}^N \pi_i &= 1 \end{aligned} \quad (4.26)$$

Typically the parameters of the models are estimated in terms of maximum likelihood using Expectation Maximization(EM). We refer the readers to (Bishop, 2007) for the detailed discussion of EM algorithm.

In our application, the number of the Gaussian components was selected as 2. This particular number is chosen due to the fact that in our case there are basically two kinds of light sources, namely arc and the weld pool. It is natural to think of the intensity distribution in each pixel (x, y) as the superposition of Gaussians. This consideration leads to the mixture of gaussian model in the form

$$I(x, y) \sim \pi_1 \mathcal{N}(\mu_1, \Sigma_1) + \pi_2 \mathcal{N}(\mu_2, \Sigma_2) \quad (4.27)$$

Both the foreground and background regions are modeled as Gaussian mixture models. The trained models are presented in Fig.4.24. In Fig.4.24 the two components of the Gaussian mixture models of the weld pool region are presented. One component situated its center as 81.5 which is corresponding to the molten pool light source, the other one 254.45 corresponding to the arc light source. This is consistent with our observation. The mixing coefficient 0.6961 approximately reflects the proportion between the area of the molten pool and the arc light.

Moreover, the train errors against the iterations during optimization using EM are also plotted in Fig.4.25. The x axis represents the iterations. The y axis represents the train error which is the mean of the negative logarithm of the probability.

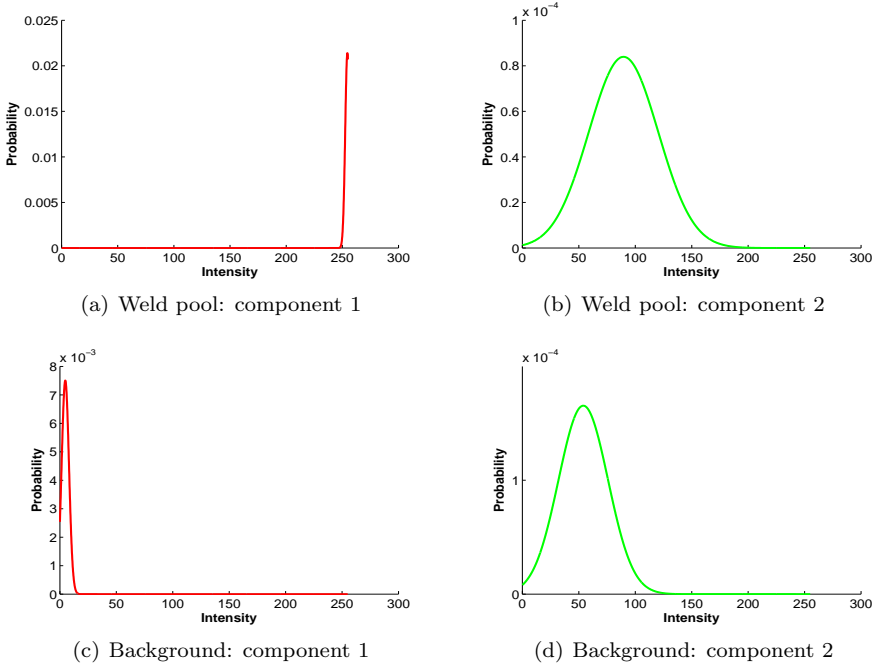
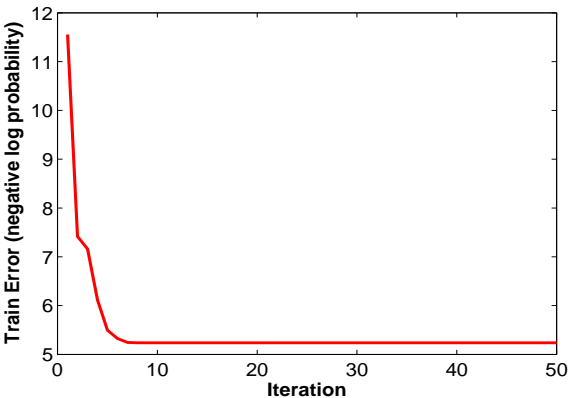
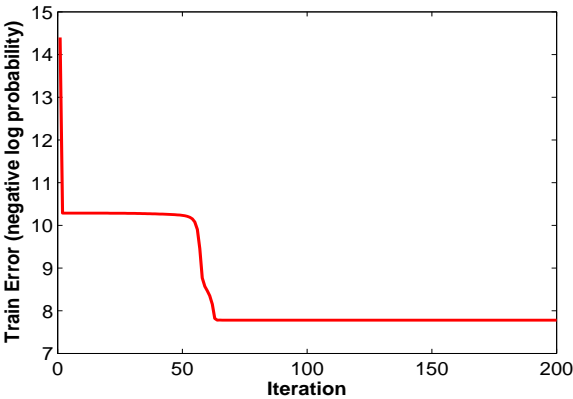


Figure 4.24: Modeling the weld pool region and the background using Gaussian Mixture Models which has two isotropic gaussian components. (a)Weld pool: $0.3054 * \mathcal{N}(254.45, 7.9^2) + 0.6946 * \mathcal{N}(81.5, 46.7^2)$. (b) Background: $0.8738 * \mathcal{N}(81.5, 46.7^2) + 0.1262 * \mathcal{N}(53.9, 31^2)$



(a) The train error of modeling the background region



(b) The train error of modeling the weld pool region

Figure 4.25: The train errors are plotted again the iterations during optimization using EM.

4.6.2 Experiments

To validate the proposed approach II, we run the algorithm on three different experiments which are the same to the ones used in validating approach I. The parameters are carefully optimized by trail and error and shown in table 4.2. Fig.4.27 shows an example that how the potential energy decreases during evolution. The potential energy here is defined as Eq.4.24 with a slight modification that the control points are classified to be either inside or outside the boundary. It is essentially a misclassification rate. It can also be seen that the minimal potential energy is reached around 200 iterations.

Table 4.2: Parameters used by Approach II

Parameters	Symbol	Value
Internal force weight : Tension	α	0.5
Internal force weight : Stiffness	β	0.1
Spacial step size	Δs	1
Temporal step size	Δt	0.5
Weight of statistical force	λ	0.05
Number of control points	N	100
Iterations	$N_{\text{iteration}}$	200

We select several typical results and shown in Fig.4.31, 4.32 and 4.33. In all the figures, the initial curves are marked in red. The final curves are marked in green. It should be noted that the initialization of approach II is different from the approach I where a periodical initialization mechanism is proposed. Here for all the frames including both arc-off and arc-on frames, the active contour starts evolving from the red circle. The center and radius of the circle is determined by computing the image moment.

4.6.2.1 Experiment 1

The results of the experiment 1 are shown in Fig.4.28 and Fig.4.31. It can be observed from both statistical and image analysis results, the performance of this approach is unsatisfactory. More specifically, it succeeded to extract the weld pool boundaries only in arc-off frames such as the frame 1, 8, 15, 17 and 18 in Fig.4.28 or as in Fig.4.31(a) and (e). For all arc-on frames, the algorithm failed.

4.6.2.2 Experiment 2

For experiment 2, the performances in arc-off frames, see Fig.4.32(a) and (c) are rather good. For some of the arc-on frames, see Fig.4.32(d), (e) and (f), the results are acceptable. For arc-on frames, see Fig.4.32(g) and (h), the algorithm failed to capture the details of the boundary and therefore unacceptable. The statistical results also agree with these observations.

4.6.2.3 Experiment 3

For experiment 3, the algorithm succeeded in capturing the dramatically changing weld pools very well with an exception of the frame 106, as shown in Fig.4.30 and Fig.4.33. The failure on the frame 106 is in fact caused by the same reason discussed in approach I, i.e., a part of the weld pool boundary is indistinct.

4.6.2.4 Speed

The algorithm is implemented in C++. To save space, we only show the speed of the algorithm in experiment 3 in Fig.4.26. The average time of processing each frame is 27.9ms(around 36Hz) on an ordinary desktop PC.

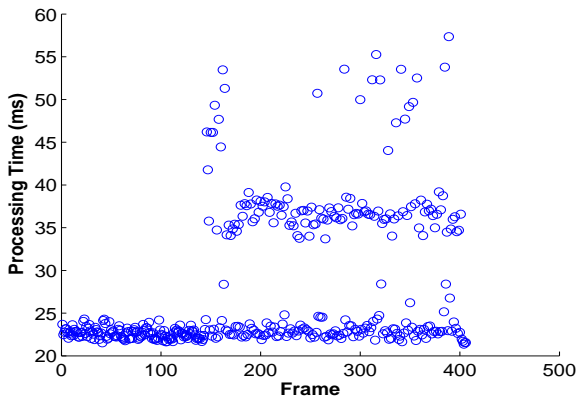


Figure 4.26: The figure shows the time of processing frames. The mean processing time of the experiment 3 is 27.9ms.

4.6.2.5 Concluding Remarks

From these results, we can see that this new active contour model does have the ability to process the arc-on frame directly and does not need a special initialization technique. In fact, the image force based on statistical models allows us to initialize the curve very easily. The accuracy of tracking the boundary is rather good in the case of having small arc light. However, for the frames with very strong arc light, the performance is limited.

Now we shall discuss when and why the proposed region-based force may fail. When deriving the active contours driven by statistical models, we make a strong assumption that the intensity of each pixel is i.i.d distributed. So the ideal case where region-based active contours perform perfectly is in fact to detect texture edges where two regions have different intensity distributions. In our applications, this is not always valid. Moreover, as we mentioned earlier, region-based forces are implemented in the framework of parametric active contours so as to be run in real time. The statistical force is transformed using Green's theorem and performs similar to a classifier or local adaptive thresholding. The consequence is that in some cases such as strong reflection exists, the active contour will be attracted by the reflection regions since they are similar to the weld pool region in terms of intensity and may fail. This actually means that in some cases, the intensity information is not sufficient to guide the evolving of active contours.

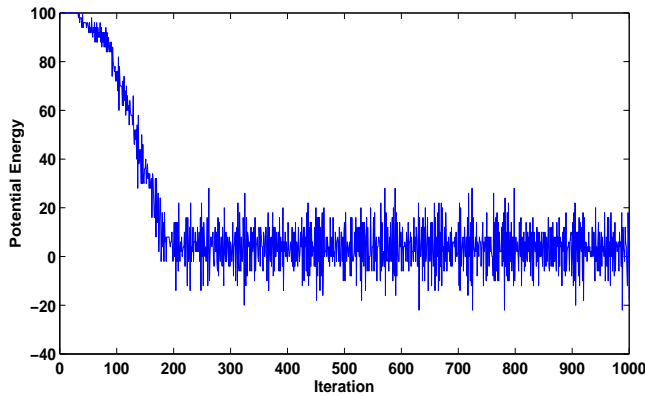
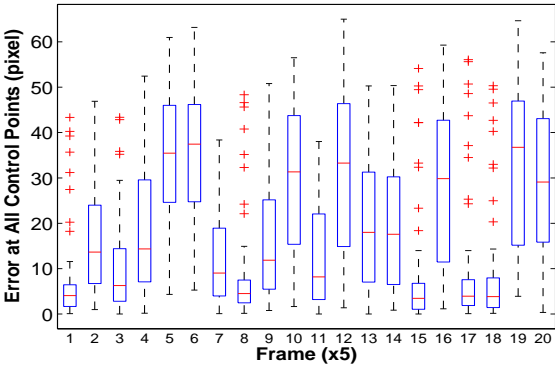
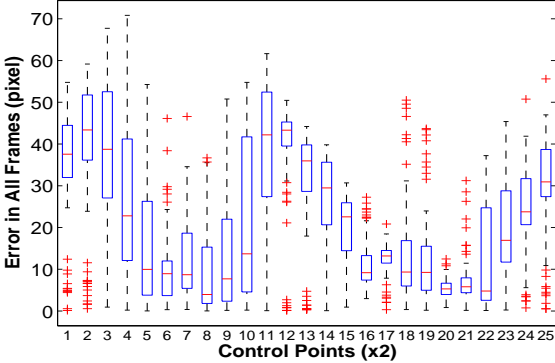


Figure 4.27: An example of minimization of the potential energy during evolution. This shows that after around 200 iterations, the curve reached an equilibrium.

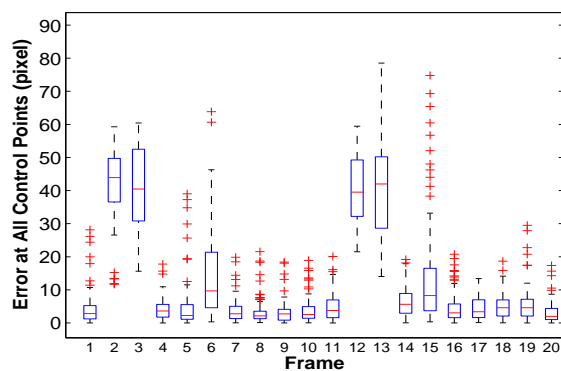


(a) Error at each frame. Note that the box plots were made every five frames

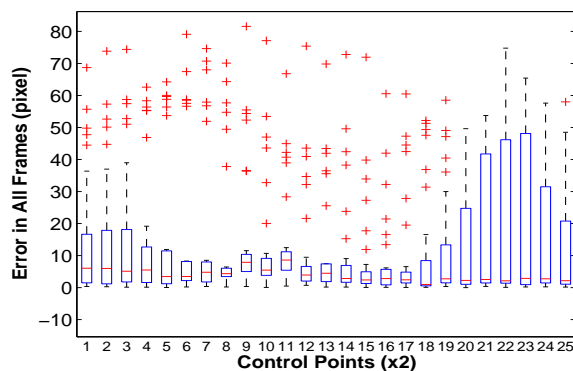


(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.28: Statistical results of testing Approach II in Experiment 1. The box plots are computed from 100 frames. See text for explanation.

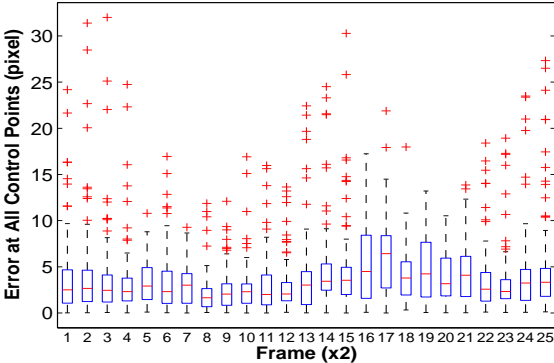


(a) Error at each frame.

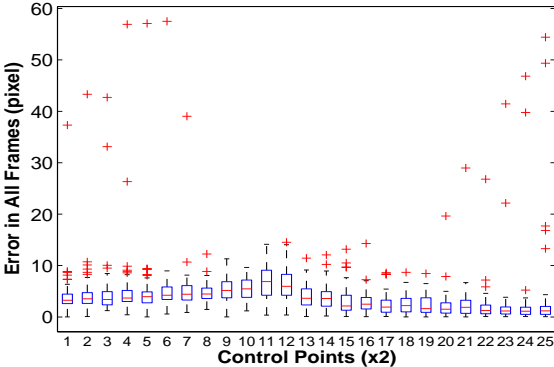


(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.29: Statistical results of testing Approach II in Experiment 2. The box plots are computed from 20 frames. See text for explanation.

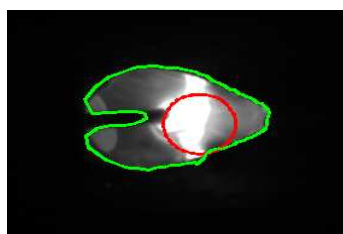


(a) Error at each frame. Note that the box plots were made every two frames

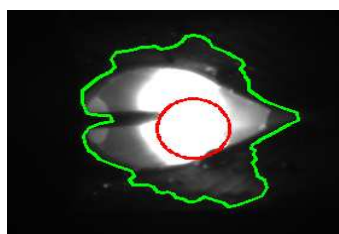


(b) Error at each control point along the boundary. The box plots were made every two control points.

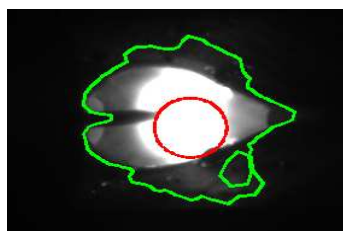
Figure 4.30: Statistical results of testing Approach II in Experiment 3. The box plots are computed from 50 frames. See text for explanation.



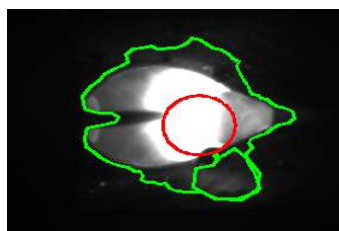
(a) Frame66



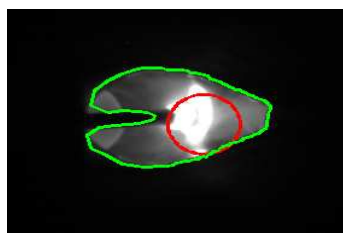
(b) Frame67



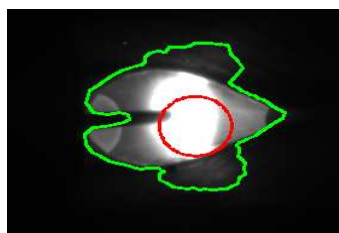
(c) Frame68



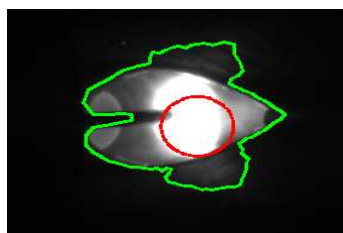
(d) Frame69



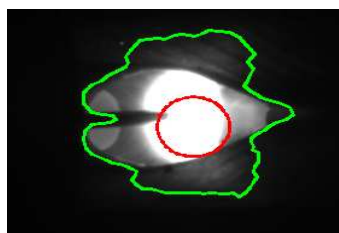
(e) Frame127



(f) Frame128

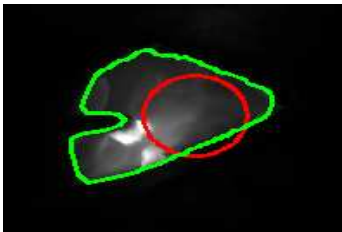


(g) Frame129

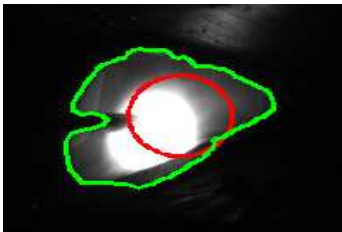


(h) Frame130

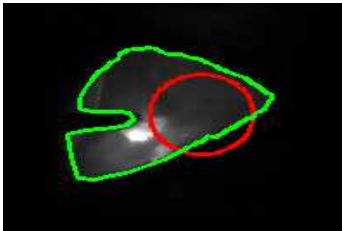
Figure 4.31: Experiment 1. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves.



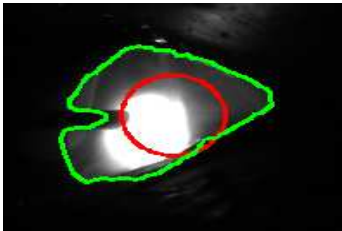
(a) Frame52



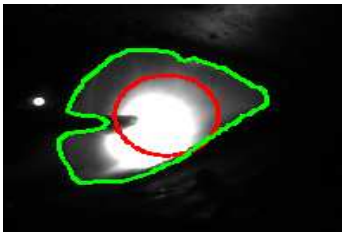
(b) Frame55



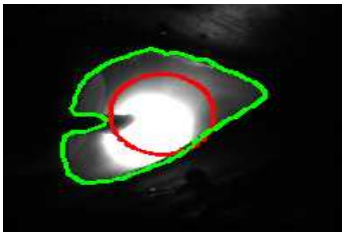
(c) Frame56



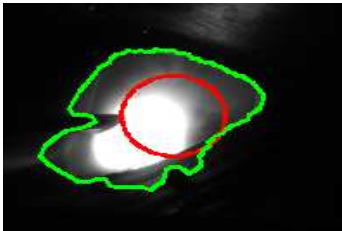
(d) Frame58



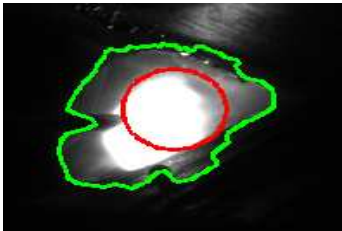
(e) Frame59



(f) Frame60

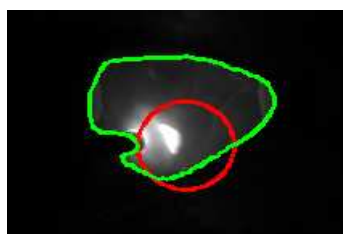


(g) Frame62



(h) Frame65

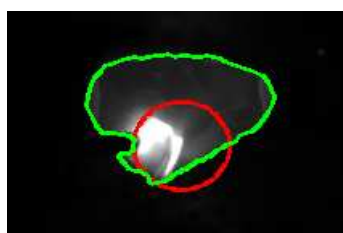
Figure 4.32: Experiment 2. In this case, the torch moved along a straight line. The extracted weld pool boundary are marked in green curves.



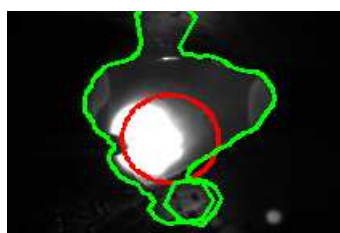
(a) Frame103



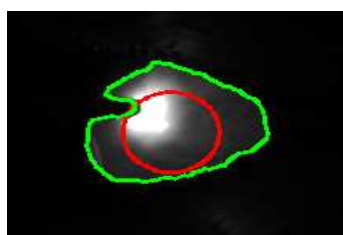
(b) Frame104



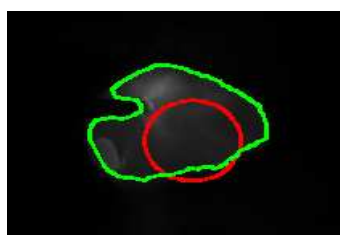
(c) Frame105



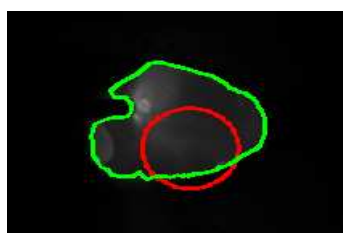
(d) Frame106



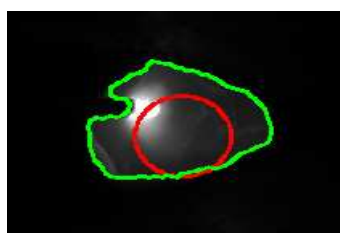
(e) Frame131



(f) Frame132



(g) Frame133



(h) Frame134

Figure 4.33: Experiment 3. Torch oscillation is required. Red circles indicate the initial positions of the active contours. The final curves are marked in green. Iterations 200.

4.7 Approach III: Feature Selective AdaBoost-ing Geodesic Active Region

4.7.1 Method

We have presented two different approaches based on active contours for weld pool tracking. Even though they have the same framework, these two active contours are actually driven by different image forces and therefore work differently.

In approach I, the image force is built based on the gradient of the original images and hence may be referred to as *boundary-based force*. The advantage is that it is simple and easy to implement and can capture the details of the weld pool boundary very well. The disadvantages are as follows: First, the initial contour has to be positioned close to the true boundary which is normally difficult. This is why we propose a periodical initialization scheme in approach I. Second, it may be attracted by false boundary and fail.

In approach II, the image force is build based on modeling different regions statistically and is referred as *region-based force*. The capture range of the active contour is greatly enlarged and both arc-off and arc-on frames can be processed directly. On the other hand, we also show that it may suffer from losing the details of the boundary.

These two approaches are not conflicting but complementary. The limitation of approach I can be partially solved by approach II and vice versa. The performance of active contours may be greatly improved by joining these two forces. In fact, much effort has been made to integrate boundary-based and region-based image segmentation techniques (Chakraborty et al., 1996; Zhu and Yuille, 1996; Paragios and Deriche, 2002a,b, 2005). In this section, we shall propose an approach which integrates both boundary-based and region-based information.

First of all, it should be pointed out the boundary information in approach I is purely gradient. It has been proved that this is not sufficient since there are edges which are defined by high gradient, but not belonging to the true boundary of the weld pool. Therefore we must investigate different kinds of boundaries and find a proper representation which can separate the true weld pool boundaries from the false ones.

4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region83

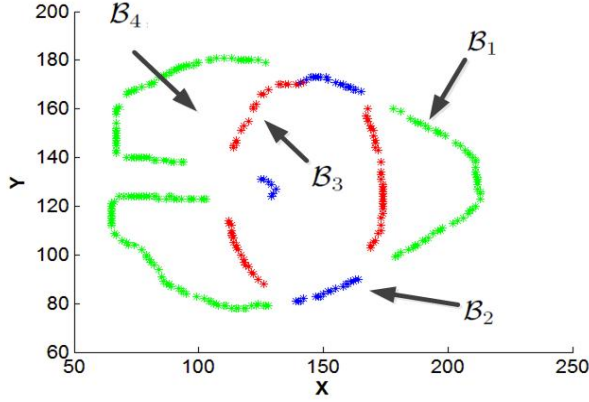


Figure 4.34: All pixels are classified into four groups: $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4$. \mathcal{B}_1 represents the boundary points being between the weld pool region and the background, marked in green. \mathcal{B}_2 represents the boundary points being between the arc light region and the background, marked in blue. \mathcal{B}_3 represents the boundary points being between the weld pool region and the arc light region, marked in red. \mathcal{B}_4 represents the non-boundary points.

4.7.1.1 Boundary Classification Using Feature Selective AdaBoost

In weld pool images, all pixels can be classified into four categories $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4$ as follows:

- \mathcal{B}_1 - the boundary between the weld pool region and the background, marked in green in Fig.4.34.
- \mathcal{B}_2 - the boundary between the arc light region and the background, marked in blue in Fig.4.34.
- \mathcal{B}_3 - the boundary between the weld pool region and the arc light region, marked in red in Fig.4.34.
- \mathcal{B}_4 - the non-boundary points.

The desired boundary that we refer to as *weld pool boundary* is composed of two classes \mathcal{B}_1 and \mathcal{B}_2 . In order to integrate these information into the tracking algorithms, we shall build a statistical model for each class, i.e., estimate the distributions $\{p_i(\mathbf{x} \in \mathcal{B}_i), i = 1, 2, 3, 4\}$, where \mathbf{x} denotes a pixel in the image.

In order to manipulate the pixel information such as building the statistical model, the pixels should be characterized in some way. For instance, in Kass's paper (Kass et al., 1988) and many other variants of active contours, the *gradient* information computed at each pixel is used to characterize the pixel. Correspondingly, the active contours in fact work on the gradient images. Another option is, such as in the Chan-Vase model (Chan and Vese, 2001), the intensity at each pixel is used. In (Olivier et al., 2008a), the authors proposed to use Haralick features to characterize the pixels to serve the purpose of extracting texture boundaries. In (Paragios and Deriche, 2002b), the authors also proposed to use some texture features to characterize the pixels. In our application, we propose to combine these information together to describe each pixel, i.e.,

$$\begin{aligned}\mathcal{F}(\mathbf{x}) &= (f_1, f_2, \dots, f_{K+4}) \\ &= (x, y, I(\mathbf{x}), G(\mathbf{x}), f_1^H, \dots, f_K^H)\end{aligned}\quad (4.28)$$

where x and y denote the position of the pixel. $I(\mathbf{x})$ denotes the intensity. $G(\mathbf{x})$ denotes the magnitude of the gradient at the pixel. $f_i^H, i = 1, 2, \dots, K$ denote Haralick features calculated at \mathbf{x} , see (Haralick et al., 1973).

The use of high-dimensional feature vector is normally sufficient, but sometime redundant, to depict the pixels. The burden of computing many features slows down the evolution of active contours and limits the real-time performance. Therefore, selecting and/or combining several components of the feature becomes necessary. Inspired by (Silapachote et al., 2005), we propose to use feature selective AdaBoost to select one or a few features which are more significant in the sense of characterizing the pixels in different classes.

The pseudo-code of the two-classes AdaBoost algorithm is shown in Algorithm 4. It can be generalized to multi-classes AdaBoost by means of *one-versus-one* strategy (Bishop, 2007). In this case, $K(K-1)/2$ binary classifiers are trained. For prediction, these classifiers ought to be combined together somehow. For instance, the prediction can be made according to a majority vote among all these binary classifiers. Or multi-class membership probabilities can be obtained by coupling the pairwise estimates of the binary classifiers and used for prediction according to Bayes theorem (Hastie and Tibshirani, 1998).

In this work Bayesian classifiers were adopted as the weak classifiers $h(\mathbf{x})$ used in AdaBoost. The reason is that Bayesian classifiers are built based upon generative probabilistic models and output class membership probability estimates. Given a sample \mathbf{x} , the output of the above AdaBoost algorithm is a series of pairwise posterior probabilities $P(\mathbf{x} \in \mathcal{B}_i | \mathbf{x} \in \mathcal{B}_i \cup \mathcal{B}_j) = p_{ij}^i$ and $P(\mathbf{x} \in \mathcal{B}_j | \mathbf{x} \in \mathcal{B}_i \cup \mathcal{B}_j) = p_{ij}^j$. These pairwise estimates given by binary classifiers can then be coupled together to obtain class membership probabilities for the overall multi-class problem $P(\mathbf{x} \in \mathcal{B}_i) = u_i, i = 1, 2, \dots, K$. The core of the

4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region85

Input: A set of hypothesis: $h_1(\mathbf{x}), \dots, h_T(x)$.

A training set: $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m)$ where $\mathbf{x}_i \in X^n$, $y_i \in \{-1, 1\}$.

Output: The final hypothesis: $H(\mathbf{x}) = \text{sign}(\sum_{t=1}^T \alpha_t h_t(\mathbf{x}))$

Initialize the weighting coefficients $D_1 = \{\omega_n, i = 1, \dots, m\}$

for $t = 1$ **to** T **do**

- a. For each feature, train a weak classifier $h_t(\mathbf{x})$ by minimizing the weighted error loss function

$$\epsilon_t = \sum_{n=1}^N \omega_n^{(t)} I(h_t(\mathbf{x}_n) \neq y_n)$$

where $I(h_t(\mathbf{x}_n) \neq y_n)$ is the indicator function and equals 1 when $h_t(\mathbf{x}_n) \neq y_n$ and -1 otherwise.

- b. Select the classifier $h_t^*(\mathbf{x})$ with the lowest error ϵ_t^* .
- c. Update the weighting coefficients as follows.

$$\alpha_t = \frac{1}{2} \ln \frac{1 - \epsilon_t}{\epsilon_t}$$

$$D_{t+1}(i) = \frac{D_t(i) \exp\{-\alpha_t y_i h_t(\mathbf{x}_i)\}}{Z_t}$$

where Z_t is a normalization factor to ensure that D_{i+1} is a distribution.

- d. Output the final model:

$$H(\mathbf{x}) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t^*(\mathbf{x})\right)$$

end

Algorithm 4: The pseudo-code of the feature selective AdaBoost.

coupling strategy is to solve a over-constrained problem described as follows:

$$\begin{cases} p_{ij}^i = \frac{u_i}{u_i + u_j} \\ p_{ij}^j = \frac{u_j}{u_i + u_j} \\ \sum u_i = 1 \end{cases} \quad i, j = 1, 2, \dots, K \quad (4.29)$$

Reordering these equation leads to

$$\begin{cases} (p_{ij}^i - 1)u_i + p_{ij}^i u_j = 0 \\ p_{ij}^j u_i + (p_{ij}^j - 1)u_j = 0 \\ \sum u_i = 1 \end{cases} \quad i, j = 1, 2, \dots, K \text{ and } i \neq j. \quad (4.30)$$

This is a linear system which has K unknowns and $K(K-1)/2 + 1$ constraints. There are more constraints than unknowns and therefore only approximated solutions can be obtained.

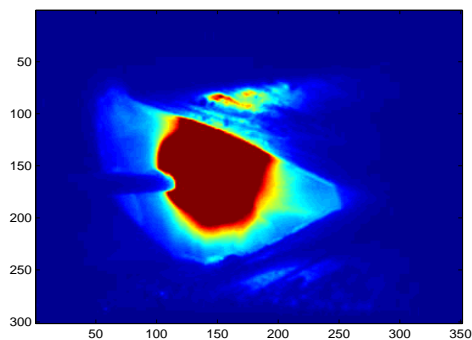
To learn the boundary classifiers, we manually label hundreds of frames. By using feature selective AdaBoost, we train Bayesian classifiers for separating each pair of the classes. Fig.4.37 shows the misclassification rate against the numbers of the weak classifiers. More classifiers, hence more features will increase the accuracy of classification. On the other hand, we must find a balance between accuracy and the amount of features. It is preferable to use less features as long as certain accuracy can be reached. By analyzing the results of AdaBoost, it is amazing that even as less as two features can reach rather accurate classification in our application. These two crucial feature are *intensity* and the Haralick feature *Contrast*. Fig.4.35 shows these two features on one typical frame. One feature is *intensity* which separates the class \mathcal{B}_1 and \mathcal{B}_2 , as well as \mathcal{B}_1 and \mathcal{B}_3 . Another one is called *Contrast* which is crucial for separating the class \mathcal{B}_2 from \mathcal{B}_3 . These results generated by AdaBoost agree our observations very well. Fig.4.36 shows graphically how boundary pixels distributed in the space spanned by these two features. It can be observed that it is possible to distinguish the pixels belonging to different boundary classes.

4.7.1.2 Unifying Boundary-based and Region-based Information

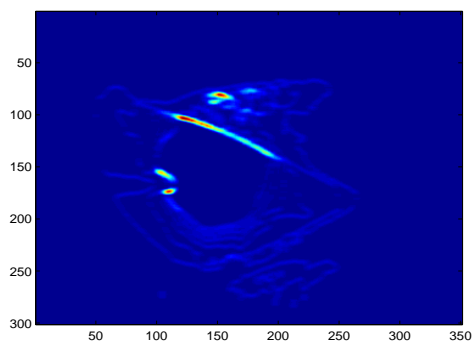
Assume that the curve is parameterized as $\mathcal{C} = (x(t), y(t)), t \in [0, 1]$. We first construct the boundary-based energy functional. According to the statistical model generated by feature selective AdaBoost, the probability that a pixel \mathbf{x} is situated on the true desired boundary is then obtained by

$$p_b(\mathbf{x}) = \begin{cases} \max\{u_1(\mathbf{x}), u_2(\mathbf{x})\} & \text{if } \mathbf{x} \in \mathcal{B}_1 \cup \mathcal{B}_2, \\ 0 & \text{if } \mathbf{x} \in \mathcal{B}_3 \cup \mathcal{B}_4. \end{cases} \quad (4.31)$$

4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region87



(a) The *intensity* which is crucial to separate the classes \mathcal{B}_1 and \mathcal{B}_2 , as well as \mathcal{B}_1 and \mathcal{B}_3 .



(b) The 2nd Haralick feature *Contrast* which is crucial to separate the classes \mathcal{B}_2 and \mathcal{B}_3 .

Figure 4.35: Crucial features selected by AdaBoost.

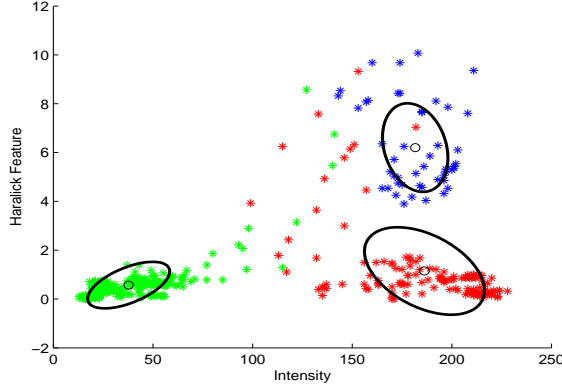


Figure 4.36: Boundary pixels in different classes described by the *intensity* and the 2nd Haralick feature *contrast*. The black ellipses indicate the two-dimensional Gaussian distribution of each class learned from labeled data.

Inspired by the work on *geodesic active regions* (Paragios and Deriche, 2002b), we also define the boundary-based energy functional as follows.

$$E_b(\mathcal{C}) = \int_0^1 g(p_b(\mathbf{x}(t))) |\dot{\mathcal{C}}(t)| dt \quad (4.32)$$

$$(4.33)$$

where $g(\cdot)$ is a monotonically decreasing function, for example,

$$g(x) = \frac{1}{1 + x^p} \quad p = 1 \text{ or } 2. \quad (4.34)$$

By exactly the same method described in approach II, the region-based information is modeled as follows.

$$E_r = - \int_{\Omega_{in}} \log \frac{p_{in}(I(\mathbf{x}))}{p_{out}(I(\mathbf{x}))} d\mathbf{x} \quad (4.35)$$

The total energy in which both the boundary and region based information are incorporated is then defined by

$$E = \eta \int_0^1 g(p_b(\mathbf{x})) |\dot{\mathcal{C}}(s)| ds - \gamma \sum_{k=1}^2 \int_{\mathcal{R}_k} \log p_r^k(I(\mathbf{x})) d\mathbf{x} \quad (4.36)$$

4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region89

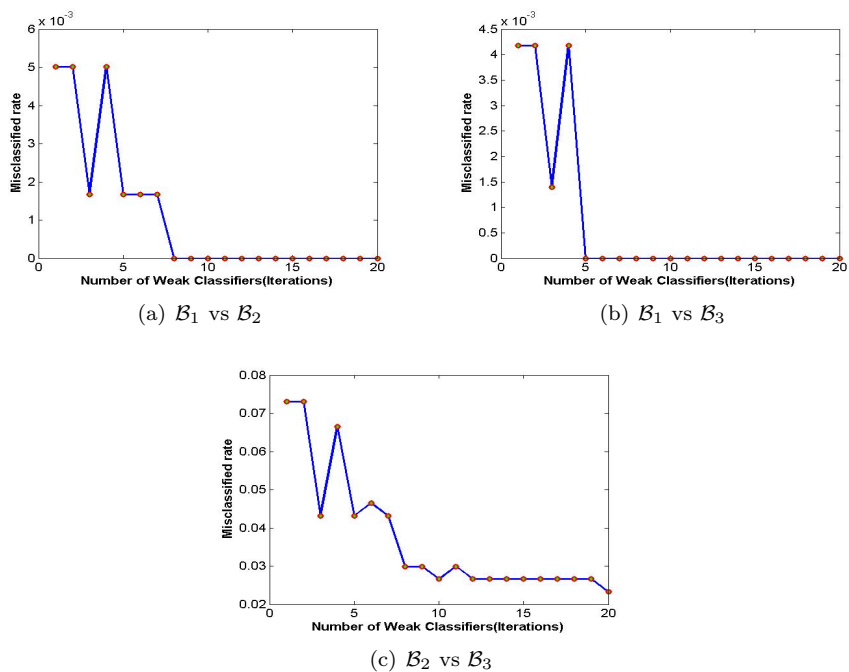


Figure 4.37: The performance of the strong classifiers against the number of the weak classifiers that were used to create the strong classifier. The x-axis represents the number of the weak classifiers. The y-axis represents the misclassification rate.

Table 4.3: Parameters used by Approach III

Parameters	Symbol	Value
Spacial step size	Δs	1
Temporal step size	Δt	0.5
Weight of boundary-based Force	η	1
Weight of region-based Force	γ	0.01
Number of control points	N	100
Iterations	$N_{\text{iteration}}$	3000

Minimizing the above energy functional by gradient decent leads to the following Euler-lagrange equation:

$$\frac{\partial \mathbf{x}(t)}{\partial t} = \eta(g(p_b(\mathbf{x}))\kappa - \nabla g(p_b(\mathbf{x})) \cdot \vec{\mathbf{n}})\vec{\mathbf{n}} - \gamma(\log \frac{p_r^1}{p_r^2})\vec{\mathbf{n}} \quad (4.37)$$

where κ denotes the curvature. $\vec{\mathbf{n}}$ denotes the unit normal vector.

4.7.2 Experiments

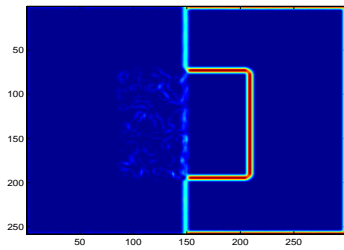
4.7.2.1 Synthetic Images

In this section, we present the validation of the proposed algorithm on both a synthetic image and the real weld pool images. The purpose of using the synthetic example is to intuitively show how the boundary-based and region-based information can be joined together to guide the evolution of active contours. It is to be noted that the feature selective AdaBoost is not required here. Fig.4.38 shows the results. The desired boundary in the center of the image partially defined by gradient and partially defined by texture. It is not a surprise that the active contour driven by only the boundary or region information fail to capture the whole boundary, as show in Fig.4.38 (b) and (c). Only if we join these two kinds of information, it succeeds, see Fig.4.38(d).

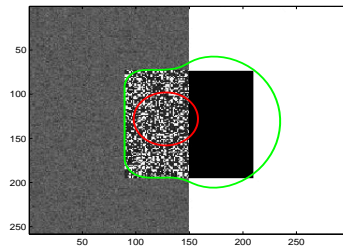
4.7.2.2 Weld Pool Tracking

Above the basic principles of the approach III FSA-GAR are presented, now we shall discuss use it in our applications. The parameters are shown in table 4.3.

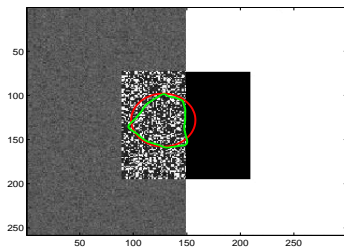
4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region91



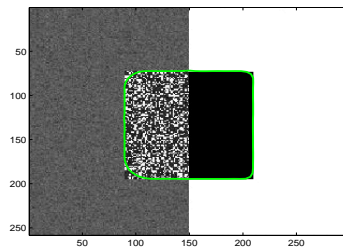
(a) The gradient



(b) The performance of an active contour driven by only region-based force.



(c) The performance of an active contour driven by only boundary-based force.



(d) The performance of an active contour driven by both the region-base and the boundary-based forces.

Figure 4.38: Testing the active contour driven by both boundary-based and region-based information on a synthetic image.

It is important to note that we make two slight improvement on the original FSA-GAR.

First, after a number of iterations using both region-based and boundary-based information, we remove the region-based information and leave only the boundary-based information to take effect so as to better capture the small details of the weld pool boundary. The approach has in fact two phases, similar to from coarse to fine procedure.

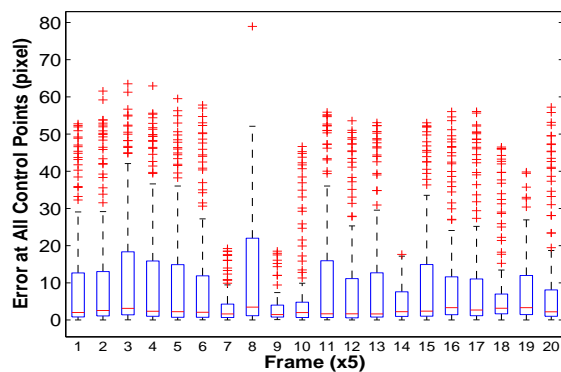
Second, one more force called *anchor* is added to facilitate active contours to form the **U** shape where the wire is located. The idea is that in our application the camera is fixed relative to the welding torch. The position of the wire in the image is therefore fixed. We put an extra constraint that if the control points of the active contour are close enough to the *anchor*, they will be captured by the *anchor* and stop moving.

Remember that FSA-GAR is proposed to overcome the drawbacks of the approach I and II, especially to tackle the problem that approach II performs unsatisfactorily when strong reflection exists. Therefore, we use experiment 1 and 2 where strong arc light and reflection appear in the images to evaluate FSA-GAR. The results can be found in Fig.4.39, Fig.4.40, Fig.4.41 and Fig.4.42.

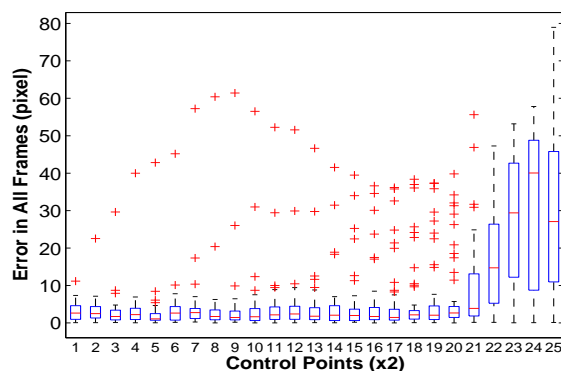
It can be seen that even with the presence of strong arc light and reflection where approach I and II fail, the algorithm succeeds to capture the details of the boundary. The performance in terms of accuracy is better than approach I and II in experiment 1 and 2. It is worth noting that in Fig.4.39 the variances of the error are large compared to such as in Fig.4.12. The reason is as follows. The approach 3 captures the welding wire boundary very well and a number of control points will be adjusted to describe the welding wire boundary which is a horizontal **U** shape. We therefore observed a significant amount of outliers in Fig.4.39(a). Moreover, it will also increase the standard deviation in each box plot. Fig.4.39(b) shows that the errors at the control points except for the ones describing the welding wire boundary are very small. This is also confirmed by Fig.4.41.

The proposed FSA-GAR is implemented in MATLAB in this stage and therefore not possible to discuss the real-time performance. We leave the implementation and test of C++ version to future work. But in theory, the speed of FSA-GAR should be comparable to the approach II since the feature selective AdaBoost only needs to be run once to find proper features as the description of each pixel.

4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region93

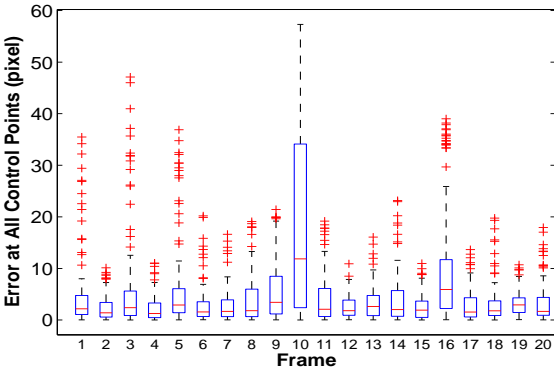


(a) Error at each frame. Note that the box plots were made every two frames

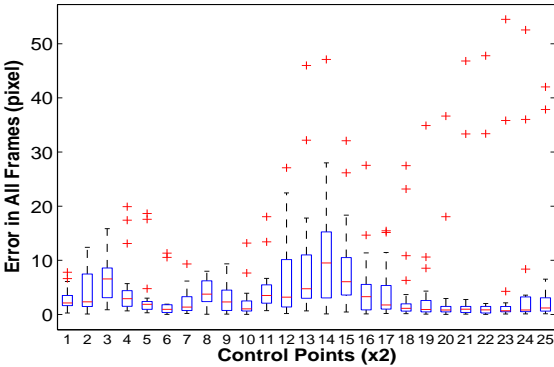


(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.39: Statistical results of testing Approach III in Experiment 1. The box plots are computed from 100 frames. See text for explanation.



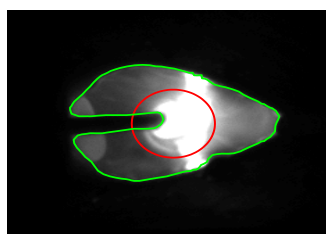
(a) Error at each frame. Note that the box plots were made every two frames



(b) Error at each control point along the boundary. The box plots were made every two control points.

Figure 4.40: Statistical results of testing Approach III in Experiment 2. The box plots are computed from 20 frames. See text for explanation.

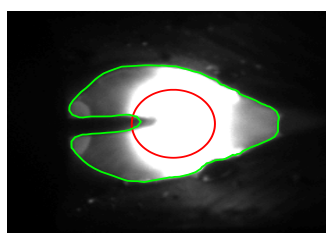
4.7 Approach III: Feature Selective AdaBoosting Geodesic Active Region



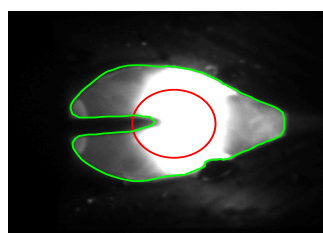
(a) Frame66



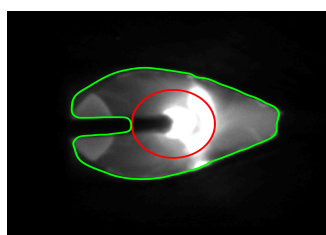
(b) Frame67



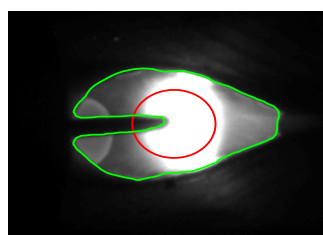
(c) Frame68



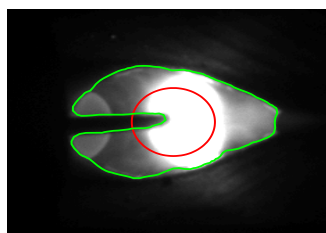
(d) Frame69



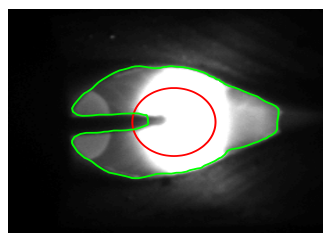
(e) Frame127



(f) Frame128

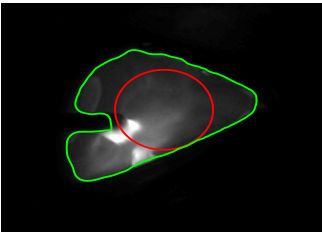


(g) Frame129

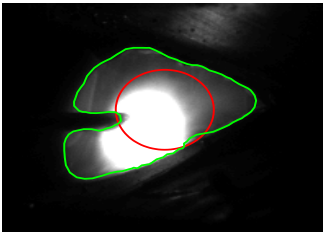


(h) Frame130

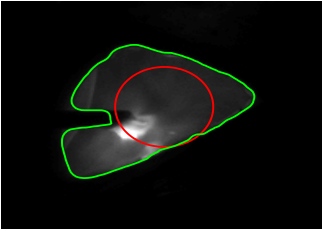
Figure 4.41: The results of using FSA-GAR on Experiment 1. The initial curve is marked in red. The final curve is marked in green.



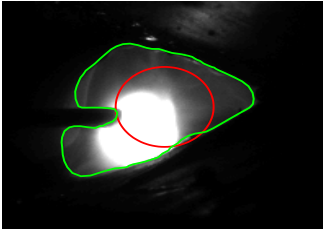
(a) Frame52



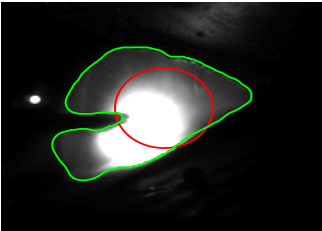
(b) Frame55



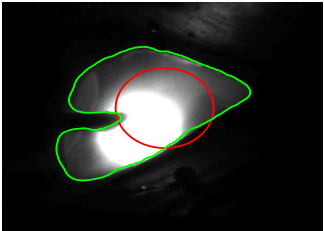
(c) Frame56



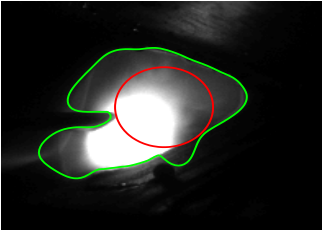
(d) Frame58



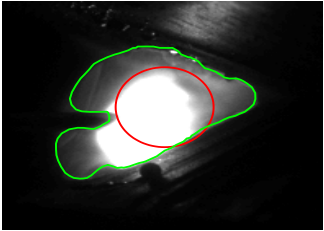
(e) Frame59



(f) Frame60



(g) Frame62



(h) Frame65

Figure 4.42: The results of using FSA-GAR on Experiment 2. The initial curve is marked in red. The final curve is marked in green.

4.8 Summary

This chapter tackles the problem of tracking the weld pool boundaries in a passive way, i.e., without using any auxiliary light source. We have presented three image analysis approaches based on parametric deformable models. A comparison is given in Table 4.4.

Table 4.4: Comparison of three weld pool boundary tracking approaches

Approach	I	II	III
Speed	12.5 ~ 50Hz	36Hz	not available
Initialization	rely on arc-off	arbitrary	arbitrary
Offline Training	no	yes	yes
Possible Welding Modes	short-circuit	all	all

The first approach is based on inflating balloons and a special initialization scheme. The frames are categorized into arc-off and arc-on frames. When an arc-off frame arrives, the image is thresholded and the weld pool boundary is extracted by an inflating balloon. The extracted weld pool boundary will also be used as the initial position of snakes for the subsequent arc-on frames. This procedure is repeated periodically. The performance of this approach has been examined in several experiments. One drawback of this approach is that its initialization relies on the arc-off frames where the arc light is not intensive. Therefore it only works in short-circuit mode. This motivates us to propose new deformable models as in the second and third approaches which are immune to the interference of the extremely strong arc light. These new deformable models no longer rely on the arc-off frames and therefore can work in different arc welding processes.

The second approach is based on active contours that are driven by region-based statistical models. By modeling different regions statistically, the capture range of active contours is greatly enlarged. It can start evolving from almost arbitrary initial positions. However, sometime it suffers from the limitation of being attracted by spurious edges. In the third approach so called feature selective AdaBoosting geodesic active regions, we further incorporate the boundary classification information into the framework of active contours. The performance of these two approaches has also been verified in several experiments.

Note that the first approach also offers a general periodical framework of using deformable models for weld pool boundary tracking. More specifically, the deformable model is initialized every certain frames. We refer to this special frame as an *initialization frame*. All the subsequent frames right after an initialization

frame will be initialized using the extracted boundary in the initialization frame. This will greatly save the computation resources. In this sense, we may replace the inflating balloons with the active contours proposed in the second and third approaches.

Finally, we end this chapter by presenting two figures where all the extracted weld pools are plotted in the image domain. The similarity between the resulting figures and the real weld beads can be observed.

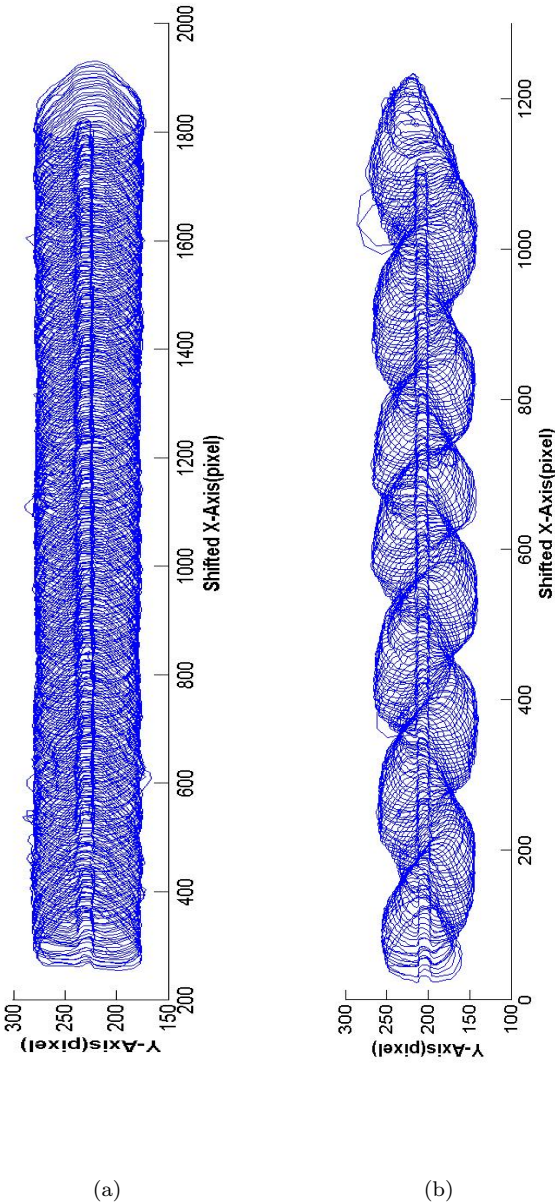


Figure 4.43: All extracted weld pools are shifted along x-axis and plotted in the image coordinate system. (a)Experiment 1 (b)Experiment 3

Towards Stabilizing Parametric Active Contours

5.1 Introduction

In chapter 4, several approaches based on active contours have been presented for weld pool boundary tracking. We select parametric implementation of active contours for real-time performance. In this chapter, we shall discuss issues regarding numerical stability of parametric active contours.

One big problem that parametric active contours often face is that the control points may bunch together or form self-intersection during evolution. The consequence is that it may lead to unreliable calculation of some quantities such as curvature, normal vector and cause the failure of the evolution. Fig.5.1 gives an example where parametric active contours failed due to this numerical instability. Numerically, this is due to the irregular spacing of control points, i.e., undesired change of the parametrization which is a side effect of parametric curve evolution. Researchers have been investigating this problem and proposed different techniques to prevent or compensate the undesired change of the parametrization (Menet et al., 1990; Cham and Cipolla, 1999; Delingette and Montagnat, 2001; Mikula and sevcovic, 2004; Srikrishnan and Chaudhuri, 2009).

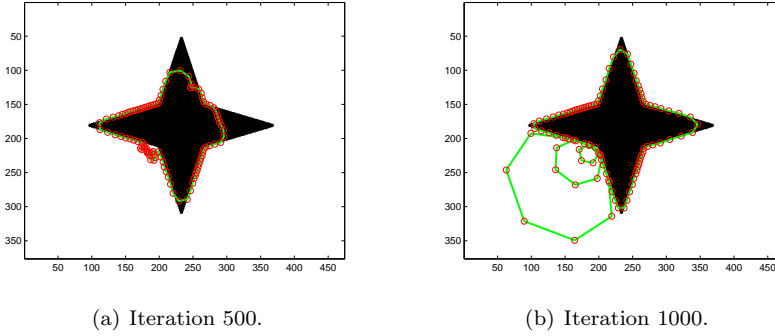


Figure 5.1: An illustration that parametric active contours can be numerical instable during evolution . (a)Control points bunched together. (b)Active contours failed.

In terms of the way, discretely or continuously, of solving this problem, these techniques can be classified into two groups. In the first group, the spacing of control points are directly and discretely controlled by re-distributing the control points. For instance in (Menet et al., 1990; Cham and Cipolla, 1999), the authors proposed to re-sample the control points every a few iterations or when the distance between neighboring points falls below a predefined threshold. This kind of techniques are straightforward and easy to implement, but also suffer from the limitation that the computation time increases due to the frequent check of the distances. Another group of techniques are such as in (Delingette and Montagnat, 2001; Mikula and sevcovic, 2004; Srikrishnan and Chaudhuri, 2009), extra tangential forces are introduced to compensate the undesired change of the parametrization during evolution.

In this chapter, a new complete tangential diffusion force will be presented for redistribution of the control points of a parametric active contour. We will show that by imposing this new complete tangential diffusion force, the change of the parametrization is subject to a diffusion equation. After iterations, the parametrization will converge to a parametrization that is proportional to the natural parametrization, i.e., the uniform distribution of the control points. We will theoretically prove that this tangential diffusion term is bounded and therefore numerical stable. Several experiments will be presented and verify the feasibility of the proposed tangential diffusion force.

5.2 Related Work

For parametric active contours, it is often important to maintain a uniform distribution of the control points along the curve during evolution. It makes the computation of geometric quantities such as the normal vector, curvature easy and stable. Assume that a 2D continuous active contour $\mathcal{C} = \mathcal{C}(x, y)$ is parameterized by q . In order to have a uniform distribution of the control points, the parametrization q should be proportional to the natural parametrization s , i.e.,

$$g(p) = \left\| \frac{\partial \mathcal{C}}{\partial p} \right\| = \zeta \quad (5.1)$$

where ζ is a constant.

For the initial curve, the control points can be uniformly distributed. However, it has been found that the parametrization may change during evolution. Discretely, the distances between neighboring points may change and are not equal. Assume that the general form of a planar curve evolution can be expressed as follows.

$$\frac{\partial \mathcal{C}(p, t)}{\partial t} = \alpha(p, t) \mathbf{T} + \beta(p, t) \mathbf{N} \quad (5.2)$$

where \mathbf{T} denotes the tangential vector. \mathbf{N} denotes the outward normal vector. $p \in [0, l]$ is a parametrization. t is the artificial evolving time. In (Kimia et al., 1992), the author proved that $g(p, t)$ evolves according to the following equation:

$$\frac{\partial g(p, t)}{\partial t} = \frac{\partial \alpha}{\partial p} + \beta \kappa g \quad (5.3)$$

where κ denotes the curvature.

It is worth noting that the tangential term α does not affect the shape of the curve but change the parametrization. The normal term β is, on the other hand, responsible for the change of the shape. Without considering the problem of numerical instability, in fact only β is essential for the curve evolution. This is the reason why only the normal term β is considered in the level set methods since one does not parameterize the curve in the level set methods. But in the parametric active contours methods, α becomes important and it is possible to actively control the evolution of the parametrization by imposing an additional tangential term.

In (Delingette and Montagnat, 2001), the authors proposed a tangential force to control the metric function $g(p, t)$.

$$f_{\text{tangent}} = \left(\frac{\partial^2 \mathcal{C}}{\partial p^2} \cdot \mathbf{T} \right) \mathbf{T} = \frac{\partial g}{\partial p} \mathbf{T} \quad (5.4)$$

By imposing this tangential force, $g(p, t)$ is govern by the following partial differential equation:

$$\frac{\partial g}{\partial t} = \frac{\partial^2 g}{\partial p^2} + \beta \kappa g \quad (5.5)$$

The above equation corresponds to the diffusion of $g(p, t)$ along the curve if β is small enough. The authors argued that when the steady state is reached, the parametrization will be proportional to the natural parametrization. However, the assumption that β is small enough is often not true in practice. The additional tangential force is insufficient to compensate all the undesired change of parametrization and therefore incomplete. The evolution of parametrization does not follow the exact heat diffusion equation. Inspired by this work, we proposed a complete tangential force that can offset all the undesired change of the parametrization.

5.3 A Complete Tangential Diffusion Force

It has been discussed that an additional tangential force $\alpha(p, t)$ can be introduced to regulate the evolution of the parametrization. Note that if α is defined as follows:

$$\frac{\partial \alpha}{\partial p} = c^2 \frac{\partial^2 g}{\partial p^2} - \beta \kappa g \quad (5.6)$$

where c is a constant. Substituting the Eq.5.6 into the Eq.5.3 leads to

$$\frac{\partial g}{\partial t} = c^2 \frac{\partial^2 g}{\partial p^2} \quad (5.7)$$

This is a *heat equation* with the periodic boundary condition $g(0, t) = g(l, t)$ and initial value $g(p, 0)$. The solution is given as follow (Asmar, 2004).

$$g(p, t) = a_0 + \sum_{n=1}^{\infty} [a_n \cos(\frac{n\pi}{l} p) + b_n \sin(\frac{n\pi}{l} p)] e^{-(\frac{cn\pi}{l})^2 t} \quad (5.8)$$

where

$$a_0 = \frac{1}{l} \int_0^l g(p, 0) dp \quad (5.9)$$

$$a_n = \frac{2}{l} \int_0^l \cos(\frac{n\pi}{l} p) g(p, 0) dp \quad (5.10)$$

$$b_n = \frac{2}{l} \int_0^l \sin(\frac{n\pi}{l} p) g(p, 0) dp \quad (5.11)$$

Let $t \rightarrow \infty$, we obtain the steady state solution $g(p, \infty) = a_0$. Note that

$$a_0 = \frac{1}{l} \int_0^l g(p, 0) dp = \frac{1}{l} \int_{\mathcal{C}_0} ds = \frac{\text{length}(\mathcal{C}_0)}{l} \quad (5.12)$$

Where $\mathcal{C}_t, t \in [0, \infty)$ denotes the curve at the time t . Very often l is set to 1, then $g(p, \infty) = \text{length}(\mathcal{C}_0)$. This proves that after sufficient iterations, the parametrization will be proportional to the natural parametrization. In other words, the control points will be distributed uniformly.

The tangential force proposed here is superior to the one in (Delingette and Montagnat, 2001) because theoretically it can offset the undesired change of the parametrization completely and therefore control the parametrization more precisely. In the following, we shall prove this both analytically and experimentally.

5.3.1 Boundness of Tangential Evolution

In this section we shall prove that the proposed tangential redistribution force $\alpha(p, t)$ is bounded i.e., $\sup_{p, t} \{\|\alpha(p, t)\|\} < \infty$ for every $p \in [0, l], t \in [t_0, \infty)$ and therefore numerically stable.

Integrating both sides of the Eq.5.6 along the curve yields

$$\alpha(p, t) = \int_0^p (c^2 \frac{\partial^2 g(p, t)}{\partial p^2} + \beta(p, t) \kappa(p, t) g(p, t)) dp \quad (5.13)$$

Hence,

$$\|\alpha(p, t)\| = \left\| \int_0^p (c^2 \frac{\partial^2 g(p, t)}{\partial p^2} + \beta(p, t) \kappa(p, t) g(p, t)) dp \right\| \quad (5.14)$$

$$\leq \int_0^l \left\| c^2 \frac{\partial^2 g(p, t)}{\partial p^2} + \beta(p, t) \kappa(p, t) g(p, t) \right\| dp \quad (5.15)$$

$$\leq \int_0^l \left\| c^2 \frac{\partial^2 g(p, t)}{\partial p^2} \right\| dp + \int_0^l \|\beta(p, t) \kappa(p, t) g(p, t)\| dp \quad (5.16)$$

$$= \int_0^l \left\| \frac{\partial g(p, t)}{\partial t} \right\| dp + \int_0^l \|\beta(p, t) \kappa(p, t) g(p, t)\| dp \quad (5.17)$$

Taking the partial derivative of both sides of the Eq.5.8 with respect to t yields

$$\frac{\partial g(p, t)}{\partial t} = \sum_{n=1}^{\infty} -\left(\frac{cn\pi}{l}\right)^2 [a_n \cos\left(\frac{n\pi}{l}p\right) + b_n \sin\left(\frac{n\pi}{l}p\right)] e^{-(\frac{cn\pi}{l})^2 t} \quad (5.18)$$

Substituting the Eq.5.18 into the first term of the Eq.5.17, we have

$$\int_0^l \left\| \frac{\partial g(p, t)}{\partial t} \right\| dp = \int_0^l \left\| \sum_{n=1}^{\infty} -\left(\frac{cn\pi}{l}\right)^2 [a_n \cos(\frac{n\pi}{l}p) + b_n \sin(\frac{n\pi}{l}p)] e^{-(\frac{cn\pi}{l})^2 t} \right\| dp \quad (5.19)$$

$$\leq \int_0^l \sum_{n=1}^{\infty} \left\| \left(\frac{cn\pi}{l}\right)^2 [a_n \cos(\frac{n\pi}{l}p) + b_n \sin(\frac{n\pi}{l}p)] e^{-(\frac{cn\pi}{l})^2 t} \right\| dp \quad (5.20)$$

$$\leq M \int_0^l \sum_{n=1}^{\infty} \left(\frac{cn\pi}{l}\right)^2 e^{-(\frac{cn\pi}{l})^2 t} dp \quad (5.21)$$

$$= Ml \sum_{n=1}^{\infty} \lambda_n^2 e^{-\lambda_n^2 t} \quad (5.22)$$

where $\lambda_n = \frac{cn\pi}{l}$ and $M = \sup_{p \in [0, l]} \{ \|a_n \cos(\frac{n\pi}{l}p) + b_n \sin(\frac{n\pi}{l}p)\| \}$. Now we shall prove that the series of functions $\sum_{n=1}^{\infty} \lambda_n^2 e^{-\lambda_n^2 t}$ converges. In fact, using *D'Alembert's ratio test*, it can be proved that the series $\sum_{n=1}^{\infty} \lambda_n^2 e^{-\lambda_n^2 t_0}$ is convergent, hence $\sum_{n=1}^{\infty} \lambda_n^2 e^{-\lambda_n^2 t}$ converges, i.e.,

$$\sum_{n=1}^{\infty} \lambda_n^2 e^{-\lambda_n^2 t} \leq \sum_{n=1}^{\infty} \lambda_n^2 e^{-\lambda_n^2 t_0} \leq W_1 \quad (5.23)$$

where W_1 is the upper bound. Therefore, the first integral of the Eq.5.17 is bounded by MLW_1 .

Now let us examine the second integral of the Eq.5.17. First, assume that $\beta(p, t)\kappa(p, t)$, which is normally defined by the need of deforming the curve so as to attach it to the desired object boundary, is bounded by F .

$$\int_0^l \|\beta(p, t)\kappa(p, t)g(p, t)\| dp \leq F \int_0^l \|g(p, t)\| dp \quad (5.24)$$

$$= F \int_0^l \left\| a_0 + \sum_{n=1}^{\infty} [a_n \cos(\frac{n\pi}{l}p) + b_n \sin(\frac{n\pi}{l}p)] e^{-(\frac{cn\pi}{l})^2 t} \right\| dp \quad (5.25)$$

$$\leq Fl|a_0| + \sum_{n=1}^{\infty} \int_0^l \left\| [a_n \cos(\frac{n\pi}{l}p) + b_n \sin(\frac{n\pi}{l}p)] e^{-(\frac{cn\pi}{l})^2 t} \right\| dp \quad (5.26)$$

$$\leq Fl|a_0| + Ml \sum_{n=1}^{\infty} e^{-(\frac{cn\pi}{l})^2 t} \quad (5.27)$$

Similarly we can prove that $\sum_{n=1}^{\infty} e^{-(\frac{cn\pi}{l})^2 t}$ converges and bounded by W_2 . Combining these two terms, it is evident to see that α is bounded, i.e.,

$$\sup_{p \in [0, l], t \in [t_0, \infty)} \left\{ \|\alpha(p, t)\| \right\} \leq Ml(W_1 + W_2) + Fl|a_0| < \infty \quad (5.28)$$

5.3.2 Convergence Speed of Tangential Diffusion

Recall that the Eq.5.8 gives the behavior of the parametrization $g(p, t)$ after adding an tangential regulating force $\alpha(p, t)$. It has been proved that $g(p, t)$ will converge to the parametrization that is proportional to the natural parametrization which indicates that the control points will be uniformly distributed. We also prove that $\alpha(p, t)$ is bounded. In practice, another important issue is to investigate the convergence speed of the proposed tangential diffusion. Again the Eq.5.8 shows that it exponentially converges to the steady-state solution and the order is given by $(\frac{cn\pi}{l})^2$. Theoretically, increasing c will speed up the convergence of the tangential diffusion. However, c can not be selected arbitrarily for the reason of numerical stability of tangential diffusion.

5.3.3 Numerical Implementation

The proposed additional tangential diffusion force α is defined as shown in the Eq.5.6. In practice, we discretize it by means of finite difference method.

$$\frac{\alpha_{i+1} - \alpha_i}{\Delta p} = c^2 \frac{g_{i+1} - 2g_i + g_{i-1}}{(\Delta p)^2} - \beta_i \kappa_i g_i \quad (5.29)$$

Reordering the terms gives us

$$\alpha_{i+1} = \alpha_i + \frac{c^2}{\Delta p} (g_{i+1} - 2g_i + g_{i-1}) - \Delta p \beta_i \kappa_i g_i \quad (5.30)$$

where

$$g_i = \left\| \frac{\mathcal{C}_{i+1} - \mathcal{C}_i}{\Delta p} \right\| \quad (5.31)$$

5.4 Experiments

For the validation of the proposed complete tangential diffusion term, three experiments have been carried out, including tests on a synthetic image, an

image of a plane and a weld pool image. The active contours used in experiment 1 and 2 are active contours driven by region-based forces. For the weld pool image, we use the proposed feature selective Adaboosting geodesic active region (FSA-GAR). The results are presented and discussed here.

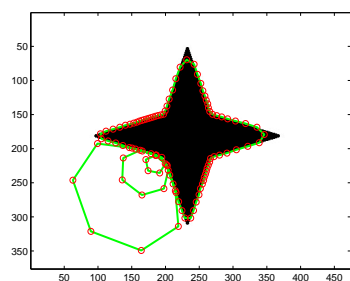
Fig.5.2 shows the results in a synthetic image. In Fig.5.2(a), the numerical instability occurred when without using any tangential force and several loops were formed and eventually failed to segment the star. On the other hand, with the proposed tangential diffusion term, it succeeded in evolving stably and capturing the boundary of the star, as shown in Fig.5.2(b). Fig.5.2(c) and (d) show the mean and standard deviation of the distance between the neighboring points.

Fig.5.3 and Fig.5.6 show another two examples in an image of a plane and a typical weld pool image. The results confirm again that the proposed tangential diffusion term can stabilize the evolution of parametric active contours.

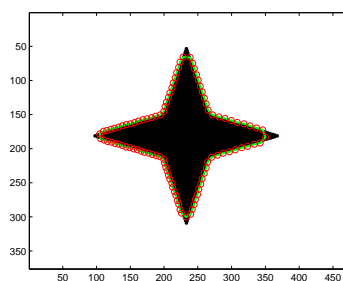
The comparison between the complete tangential force proposed here and the incomplete one proposed in (Delingette and Montagnat, 2001) has also been carried out. The results are presented in Fig.5.4 and Fig.5.5. It can be seen that with the complete tangential force, the control points are much more evenly distributed.

5.5 Summary

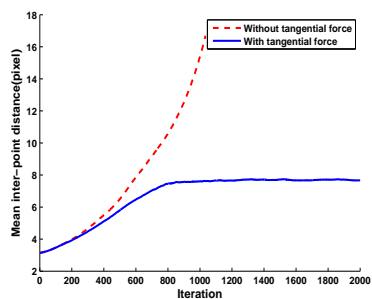
Parametric active contours often face the problem of numerical instability during evolution which is mainly caused by undesired change of the parametrization. In this chapter, a new tangential diffusion term is proposed to compensate this undesired change. We analytically prove that by adding this term, the parametrization will converge to the one that is proportional to the natural parametrization. This implies that the control points are distributed uniformly. The experiments in synthetic and real-world images verify the feasibility of the proposed tangential diffusion term.



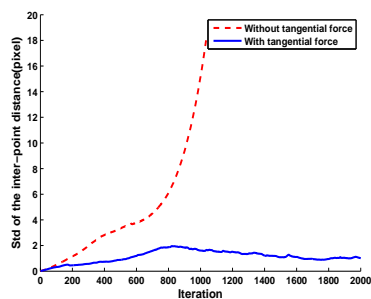
(a) Without tangential forces.



(b) With the proposed tangential force.



(c) Mean of the inter-point distance.

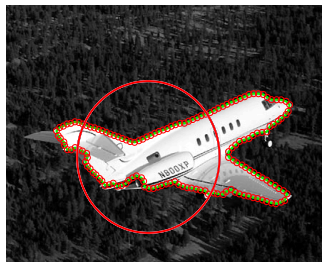


(d) Standard deviation of the inter-point distance.

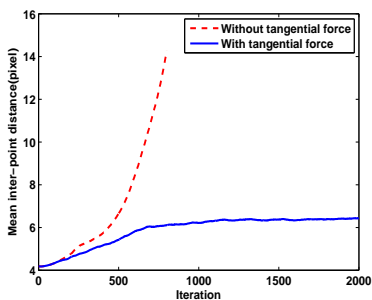
Figure 5.2: Comparison of curve evolution with or without the proposed tangential force in a synthetic image.



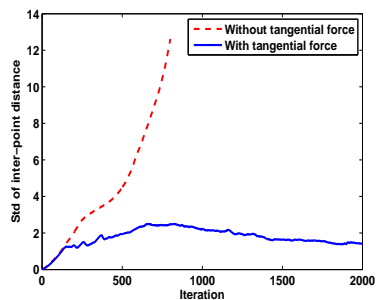
(a) without tangential force.



(b) with tangential force.



(c) Mean of the inter-point distance.



(d) Standard deviation of the inter-point distance.

Figure 5.3: Comparison of curve evolution with or without the proposed tangential force in an image of a plane. The red circles indicate the initial contours. The final curve are marked in green.

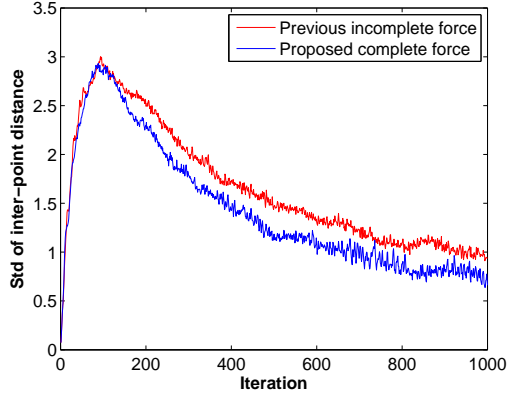


Figure 5.4: Comparison of the proposed complete tangential force and the previous incomplete tangential force in the plane image. The x-axis represents the iteration. The y-axis represents the standard deviation of the distance between neighboring points.

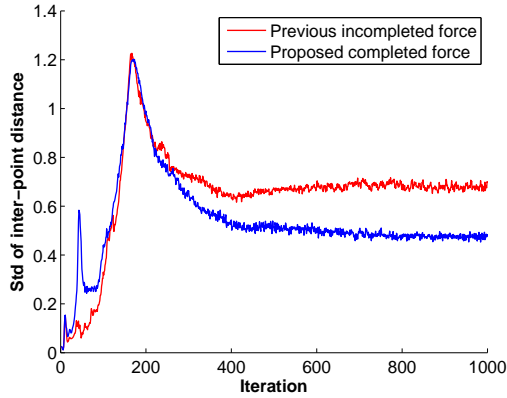
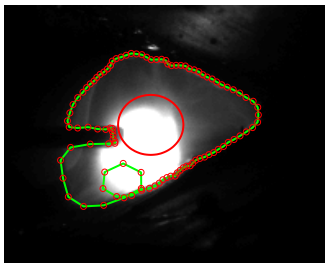
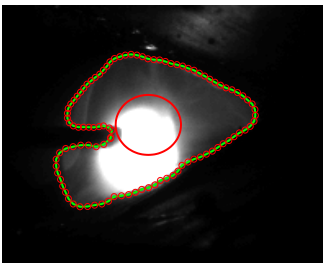


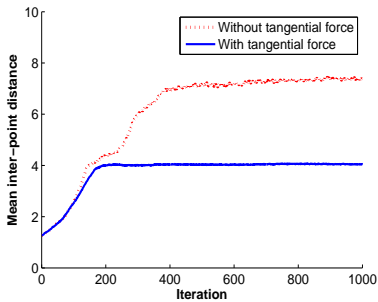
Figure 5.5: Comparison of the proposed complete tangential force and the previous incomplete tangential force in the weld pool image. The x-axis represents the iteration. The y-axis represents the standard deviation of the distance between neighboring points.



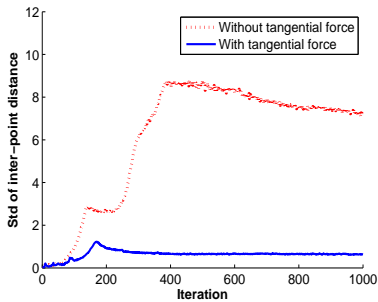
(a) without tangential force.



(b) with tangential force.



(c) Mean of the inter-point distance.



(d) Standard deviation of the inter-point distance.

Figure 5.6: Comparison of curve evolution with or without the proposed tangential force in a weld pool image. The red circles indicate the initial contours. The final curve are marked in green.

Vision-based Control

6.1 Introduction

In the previous chapters, we have proposed a passive vision system which is capable of sensing the weld pool and measuring the seam profile simultaneously in arc welding. It has been mentioned that the purpose of extracting this visual information is to form a closed-loop control system based on this information so as to improve the productivity and welding quality. In this chapter, we address the issue of controlling the welding robot and welding process based on the extracted visual information.

It is important to note that two major tasks involving closed-loop control here should be distinguished from each other. One is so-called *seam tracking*. With this function, the control system should be able to navigate the welding torch to follow the seam or form a special pattern of the joint. The other is often referred to as *adaptive control of the welding condition* involving the adjustment of welding parameters in a weld. Both of these two tasks may require the information of the seam and/or the weld pool. From the perspective of control schemes, The former task, i.e., seam tracking mainly falls into the category of vision-based robot control, or so called visual servoing. The latter task is a conventional control problem, except that the feedback information is extracted by visual sensors.

In this chapter, one case study regarding seam tracking will be presented and discussed, and can also serve as a further evaluation of the proposed vision system. We have developed a closed-loop seam tracking system which utilizes the seam profile information as the feedback to the control system. In this application, the inner width of the seam is rather small and torch oscillation is not involved. The control objective is to drive the welding torch to follow the center line of the seam to make a quality root path welding.

The remainder of this chapter is organized as follows. In Section 6.2, we make a brief introduction to visual servoing. In Section 6.3, we discuss specific control strategies for a seam tracking system which makes use of the passive seam detection algorithm presented in Chapter 3. In Section 6.4, we demonstrate the passive seam tracking system by presenting the results from real welding experiments. In the end, Section 6.5 summarizes this chapter.

6.2 Visual Servoing: A Brief Introduction

Vision-based robot control, or visual servoing, is concerned with the problem of controlling a robot using the information provided by a vision system. In terms of the error used to generate the control law, vision-based robot control can be classified as follows: *position-based visual servoing*, *image-based visual servoing* etc. Here we shall make a very brief introduction. For more detailed discussions, we refer the readers to survey papers (Malis et al., 1999; Malis, 2002).

In Position-Based Visual Servoing(PBVS), the desired pose of the camera is given in the world coordinate system. If the current pose of the camera can be estimated from the current images, it is then possible to generate the control law that uses the error between the current and the desired pose of the camera. Since the error is often computed in 3D Cartesian space, it is often called *3D visual servoing*. The main disadvantage of 3D visual servoing is that it requires accurate image feature extraction and accurate camera calibration for achieving a good performance. However, both of these two are usually difficult to be obtained. Another drawback is that in position-based methods, no explicit control in the image exists and hence the target may leave the camera field of view and causes a failure as a result. Fig. 6.1 shows the diagram of PBVS.

On the contrary, in Image-Based Visual Servoing(IBVS), the closed-loop control is performed in the 2D image space instead of the 3D world space. Errors are typically computed by comparing the current visual features with the reference visual features when the robot reaches its desired pose. This approach is often called *2D visual servoing* since the control is carried out in 2D image space.

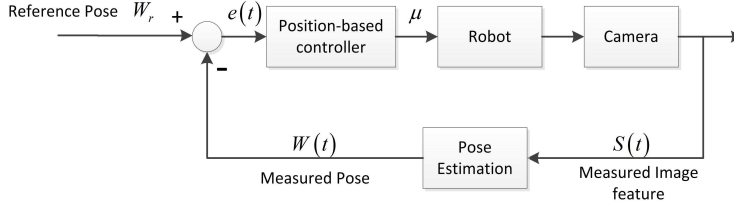


Figure 6.1: Diagram of position-based vision servoing.

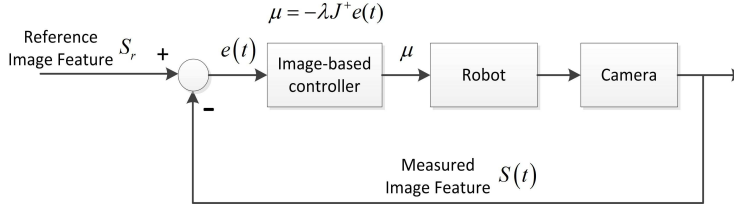


Figure 6.2: Diagram of image-based vision servoing.

Moreover, it avoids the accurate camera calibration and possesses the advantage of being robust with respect to the camera and robot calibration errors. An important concept here is so-called *image Jacobian* which relates the displacement of image features to the camera motion. It allows us to calculate the movement of the camera, create control commands according to the dynamic of the image features and hence perform a closed-loop control. The drawback of 2D visual servoing is that only in a neighborhood of the desired position, the convergence is ensured. The control diagram of IBVS is shown in Fig. 6.2 .

6.3 Visual Servoing for Seam Tracking

Even though the main goal of this PhD project is to develop a vision system for arc welding robots, it is still important to discuss and investigate the design of the control strategies based on the extracted visual features. In the previous chapters, the proposed vision system is mainly evaluated in terms of image analysis results. By closing the loop of control, we are able to evaluate the proposed vision system in terms of control and welding.

In our experiment setup, a 3 degree-of-freedom(dof) manipulator (X, Y, Z) was used in all of our experiments. Both the welding torch and the camera are mounted to the manipulator. The configuration of the whole system is then

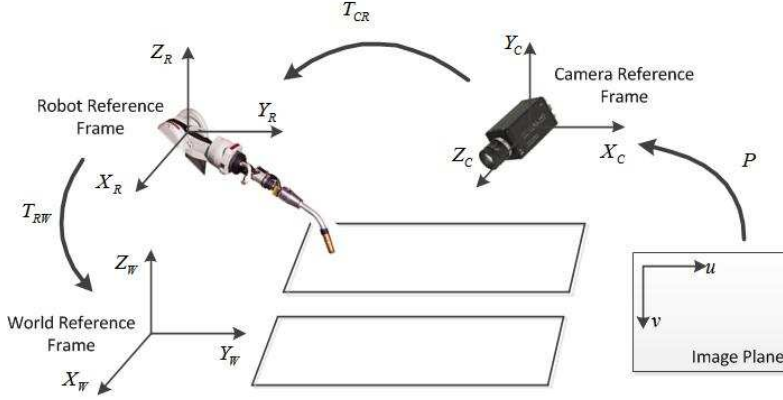


Figure 6.3: The world, robot and camera reference frames.

eye-in-hand. The controlled variable is the position (x, y, z) of the welding torch.

For the purpose of simplicity, meanwhile without sacrificing the effectiveness of validating the proposed vision system, the movement of the welding torch is limited within the plane (X, Y) where the workpiece lies in. In the next few paragraphs, we shall discuss the design of control strategies for seam tracking, including position-based and image-based visual servoing.

Before considering the control strategies, the involved coordinate systems and corresponding transformations shall be discussed. Fig. 6.3 gives a graphically illustration. In general, there are four coordinate systems involved: the image coordinate system $\{U, V\}$; the camera coordinate system $\{X_C, Y_C, Z_C\}$ the origin of which is at the camera center and the axes X_C and Y_C are parallel to the image axes U and V respectively; the robot coordinate frame $\{X_R, Y_R, Z_R\}$ attached to the robot; the world coordinate frame $\{X_W, Y_W, Z_W\}$ that is fixed to the environment. In our experiment setup, the camera is fixed with respect to the welding torch. For convenience, the robot coordinate system can be positioned so as to be identical to the camera coordinate system.

6.3.1 Position-based Visual Servoing

Camera Calibration

Assume that the center line of the seam is extracted in the image coordinate system as

$$au + bv + c = 0 \quad (6.1)$$

Recall that the movement of the torch is confined within the $\{X, Y\}$ plane. The map from a point in the image plane $\{U, V\}$ to the corresponding point in the world plane $\{X, Y\}$ can be depicted by a homography matrix H . In fact, this can be considered as a partial calibration of the camera. We are then able to compute the center line of the seam in the world coordinate system as follows:

$$\begin{bmatrix} a' \\ b' \\ c' \end{bmatrix} = H^{-T} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (6.2)$$

where the center line of the seam in the world coordinate system is represented as

$$a'x + b'y + c' = 0 \quad (6.3)$$

Robot Control

The objective of seam tracking is to make the welding torch moving along the center line as shown in Eq. (6.3) with a constant or varying speed which is determined by the seam and the welding process. From the control perspective, the control object here is the position of the welding torch $\mathbf{P}_t = (x_w, y_w)$. The control input is the velocity of the welding torch $\mathbf{V}_t = (v_x, v_y)$. The error is defined as the distance from the welding torch to the center line of the seam, as shown in Fig. 6.4. To reduce the error, a PID-based controller is employed to control the movement of the welding torch.

A few remarks should be made here. Firstly, the PBVS system that we proposed to use here is a special version in the sense that the robot space is limited to 2D, not typical 3D. This offers us the advantage of greatly simplifying the task of camera calibration: only a homography matrix from the image plane to the robot $\{X, Y\}$ plane, instead of the intrinsic and extrinsic parameters of the camera, is needed to be estimated. Secondly, the downside of this approach, as most PBVS systems have, is that it requires accurate camera calibration, i.e., estimating the homography matrix.

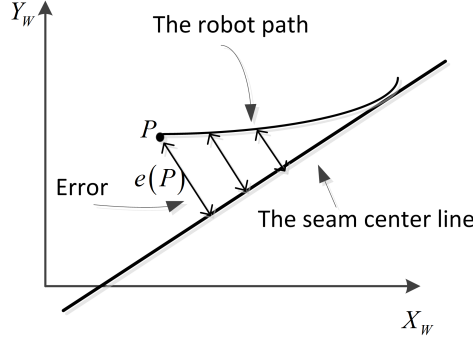


Figure 6.4: The definition of the error of the robot following a straight line seam.

6.3.2 Image-based Visual Servoing

Visual Features

In the framework of IBVS, the error of visual features is computed in the image domain. So we shall define or select k adequate visual features to control m DOF.

Image Jacobian

As discussed previously, an image-based visual servoing system computes the feedback error $e(s)$ in the image space. As depicted in Fig. 6.2, f_r denotes the reference image feature, $f(t)$ denotes the measured image feature. $e(s)$ is defined as follows.

$$e(\mathbf{s}) = \mathbf{f}_r - \mathbf{f}(t) \quad (6.4)$$

On the other hand, the adjustment of the robot position demands control inputs in either robot joint coordinates or task space coordinates. Therefore it is necessary to derive the relationship between changes to the image features and changes to the pose of the robot. In fact, the relationship can be depicted by the following equation.

$$\dot{\mathbf{f}} = J_v \dot{\mathbf{r}} \quad (6.5)$$

where $\dot{\mathbf{f}}$ denotes the velocity of a feature point in the image space, $\dot{\mathbf{r}}$ denotes the velocity screw of the corresponding point in the Cartesian space. J_v is so-called

image Jacobian.

$$J_v(\mathbf{r}) = \left\{ \frac{\partial \mathbf{f}}{\partial \mathbf{r}} \right\} = \begin{bmatrix} \frac{\partial f_1}{\partial r_1} & \cdots & \frac{\partial f_1}{\partial r_m} \\ \vdots & \cdots & \vdots \\ \frac{\partial f_k}{\partial r_1} & \cdots & \frac{\partial f_k}{\partial r_m} \end{bmatrix} \quad (6.6)$$

In practice, image Jacobian matrices can be deduced analytically via calibrating the camera system or be estimated numerically.

To generate control laws, the inverse of an image Jacobian matrix is of our interest. Assume that J_v is nonsingular and invertible, J_v^{-1} exists and the following equation holds

$$\dot{\mathbf{r}} = J_v^{-1} \dot{\mathbf{f}} \quad (6.7)$$

When J_v^{-1} does not exist, we can use an appropriate pseudo-inverse such as

$$J_v^+ = (J_v^T J_v)^{-1} J_v^T \quad (6.8)$$

In this case, we then have

$$\dot{\mathbf{r}} = J_v^+ \dot{\mathbf{f}} \quad (6.9)$$

It is to be noted that when J_v is nonsingular, J_v^+ degenerates to the regular inverse J_v^{-1} .

Now we shall derive the image Jacobian in our experiment setup. Suppose that a point in image space is used as the visual feature, denoted as $\mathbf{f} = (u, v)$ in the image coordinate system. The corresponding 3D point rigidly attached in the end-effector is denoted as $\mathbf{r}_c = (x_c, y_c, z_c)$ in the camera coordinate system.

Using the pinhole camera model, we have

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim P_{3 \times 4} \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \quad (6.10)$$

where P is the camera matrix and the points are presented in homogeneous coordinates.

$$P = \begin{bmatrix} f s_x & 0 & u_0 & 0 \\ 0 & f s_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (6.11)$$

f is the focal length of the camera. s_x and s_y are the densities of pixels along u and v (e.g. in number of pixels per mm). (u_0, v_0) is the principal point in image plane.

Taking the derivative of both sides of Eq. (6.10) with respect to time t leads to

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} \frac{fs_x}{z_c} & 0 & -\frac{x_c}{z_c^2} \\ 0 & \frac{fs_y}{z_c} & -\frac{y_c}{z_c^2} \end{bmatrix} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{z}_c \end{bmatrix} \quad (6.12)$$

or equivalently

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} \frac{fs_x}{z_c} & 0 & -\frac{u-u_0}{z_c} \\ 0 & \frac{fs_y}{z_c} & -\frac{v-v_0}{z_c} \end{bmatrix} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{z}_c \end{bmatrix} \quad (6.13)$$

where (\dot{u}, \dot{v}) represents the velocity of the image feature point, i.e., optical flow in the image plane. $(\dot{x}_c, \dot{y}_c, \dot{z}_c)$ represents the velocity of the corresponding 3D point in the camera coordinate system. Assume that the transformation from the camera coordinate system to the world coordinate system is $[\mathbf{R}_{cw}, \mathbf{t}_{cw}]$. The camera coordinate system can be rotated by applying \mathbf{R}_{cw} to be aligned to the world coordinate system. In other words, the new coordinate system differs from the world coordinate system only in terms of a translation. The coordinates of the 3D point in this rotated coordinate system are denoted by $\mathbf{r}_{c_r} = (x_{c_r}, y_{c_r}, z_{c_r})$.

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \mathbf{R}_{cw} \begin{bmatrix} x_{c_r} \\ y_{c_r} \\ z_{c_r} \end{bmatrix} \quad (6.14)$$

Combining Eq. 6.13 and 6.14, we have

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} \frac{fs_x}{z_c} & 0 & -\frac{u-u_0}{z_c} \\ 0 & \frac{fs_y}{z_c} & -\frac{v-v_0}{z_c} \end{bmatrix} \mathbf{R}_{cw} \begin{bmatrix} \dot{x}_{c_r} \\ \dot{y}_{c_r} \\ \dot{z}_{c_r} \end{bmatrix} \quad (6.15)$$

In our experiment setup, it can be seen that the velocity of the camera with respect to the world coordinate system is equal to the velocity of the 3D point in the workspace with respect to the rotated camera coordinate system, except the opposite orientations. The image Jacobian is therefore given by

$$J_v = \begin{bmatrix} \frac{fs_x}{z_c} & 0 & -\frac{x_c}{z_c^2} \\ 0 & \frac{fs_y}{z_c} & -\frac{y_c}{z_c^2} \end{bmatrix} \mathbf{R}_{cw} \quad (6.16)$$

To generate control laws, it is often necessary to invert the image Jacobian matrix. J_v as stated in Eq. (6.16) is a 2×3 matrix and is therefore not invertible. In this case, normally we shall use the pseudo-inverse of an image Jacobian matrix.

It is important to note that if only one image feature point is used to construct the image Jacobian matrix, such as J_v as stated in Eq. (6.16), an appropriate

pseudo-inverse, assume that J_v is full rank, is given by (Hutchinson et al., 1996)

$$J_v^+ = J_v^T (J_v J_v^T)^{-1} \quad (6.17)$$

If more than one image feature points are used to construct the image Jacobian matrix. The typical way is to stack all the image Jacobian together to obtain an over-estimated matrix. In this case, an appropriate pseudo-inverse is then given as follows.

$$J_v^+ = (J_v^T J_v)^{-1} J_v^T \quad (6.18)$$

Resolved-Rate Motion Control

Resolved rate motion control, first proposed by Whitney, is a common technique for robotic control (Whitney, 1972). Following this control scheme, the control input is often defined to be related to the velocity of the end-effector with respect to the world coordinate system. It is important to note that in our experiment setup, this velocity is negatively equal to $\dot{\mathbf{r}}_c$ which is the velocity of a 3D point on the workpiece with respect to the camera coordinate system.

$$\mathbf{u} = -\dot{\mathbf{r}}_c \quad (6.19)$$

According to Eq. (6.9), we have

$$\dot{\mathbf{r}}_c = J_v^+ \dot{\mathbf{f}} \quad (6.20)$$

Inserting Eq. (6.20) into Eq. (6.19) leads to

$$\mathbf{u} = -J_v^+ \dot{\mathbf{f}} \quad (6.21)$$

A proportional control law is then given by

$$\mathbf{u} = -k J_v^+ e(\mathbf{f}) \quad (6.22)$$

where $e(\mathbf{f})$ is the error function as defined in Eq. (6.4).

6.4 Experiments

This section presents the welding results of using a closed-loop seam tracker. The closed-loop seam tracker is based on the passive seam detection algorithm presented in Chapter 3. The control strategy is the position-based visual servoing strategy discussed above with a proportional controller.

Fig.6.5 shows a comparison between the actual robot path and the true center line of the seam. The robot path is marked in blue. The true center line of the seam is marked in red. The error of the actual robot path is also plotted in Fig.6.6. It can be seen that the robot follows the seam well.

Fig.6.7 show a comparison of the welding results with and without feedback control by using the passive seam tracking system. It can be seen that the welding robot can follow the seam very well by making use of the passive seam tracking system. If without using any seam tracking system, the offset is so significant and results in a failure.

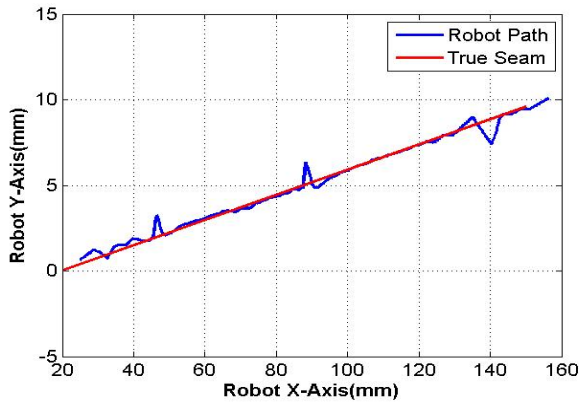


Figure 6.5: A comparison between the actual robot path and the true center line of the seam. The robot path is marked in blue. The true center line of the seam is marked in red. See text for more explanation.

6.5 Summary

This chapter discusses how to incorporate the visual information obtained by a vision system into the control system so that the control loop can be closed using the visual input. We have discussed the control strategy for implementing a closed-loop seam tracker by using the proposed passive seam detection system. The experiment has been carried out and the results verify the feasibility of the proposed closed-loop seam tracker.

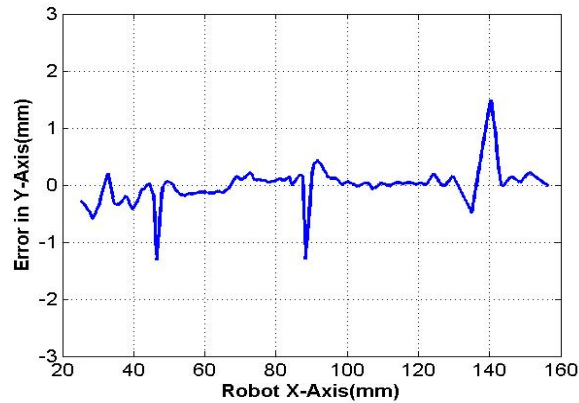


Figure 6.6: The residual error of the robot path compared to the true center line of the seam.

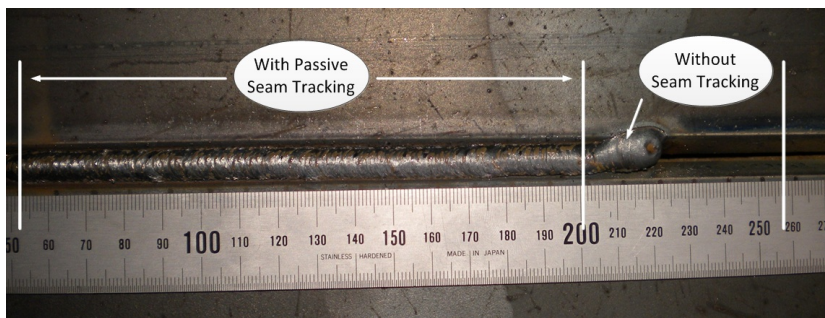


Figure 6.7: A comparison of the welding results with and without feedback control by using the passive seam tracking system. It can be seen that the welding robot can follow the seam very well by making use of the passive seam tracking system. If, however, without using any seam tracking system, the offset is so significant and results in a failure.

Conclusions and Future Work

In this dissertation, we have presented the investigation of developing a passive vision system which is able to capture the same information as we believe are used by well-trained welders. The system is only equipped with a single off-the-shelf CCD camera and a carefully-selected narrow bandpass filter, yet capable of detecting the seam geometry and the weld pool boundary simultaneously in real time. This has been achieved by means of sophisticated image analysis and machine learning techniques.

As discussed in Introduction 1, one of the principle concerns in this project is to seek image analysis techniques for detecting and representing non-rigid weld pools. We have shown that deformable models can be the answer. More specifically, three different weld pool boundary tracking approaches based on deformable models have been proposed. Experiments have verified the feasibility and effectiveness of these approaches.

The remainder of this chapter will recapitulate the technical contributions we have achieved and outlines a few directions for future work.

7.1 Contributions

The dissertation includes the following significant contributions to the field of arc welding automation and image analysis:

1. We have developed a complete passive vision system for arc welding automation. The system is only equipped with a single CCD camera and a narrow bandpass filter, yet capable of detecting the seam and the weld pool boundary simultaneously. Ability to obtain both the seam and the weld pool information simultaneously is a great advantage for developing closed-loop control systems of the welding process. For instance, as discussed in Chapter 3, it will allow us to establish a closed-loop seam tracker which directly controls the weld pool position and shape, instead of the welding torch position. (Chapter 2 and the subsequent chapters)
2. We have designed a passive seam detection algorithm which is capable of measuring the seam profile very close to the weld pool region. It is superior to the popular laser scanner solution which can only measure the seam at a distance ahead to the weld pool regions. (Chapter 3, published in JOM16 (Liu et al., 2011a))
3. We have designed three different approaches based on parametric active contours for weld pool boundary tracking. The first approach is based on inflating balloons and a special periodical initialization strategy. This approach requires periodically re-initialization from frames showing very little arc light. It is suitable for the case of short-circuit arc welding, but may perform unsatisfactory if the arc light appears all the time. (Chapter 4, Section 4.5, published in IEEE CASE2011 (Liu et al., 2011b).)
4. To tackle the initialization problem shown in the first approach, we have proposed a second approach which is a region-based active contour driven by statistical models of the image intensity in different regions. This approach has the ability to start evolving from an almost arbitrary initialization with or without the presence of much arc light and is therefore expected to be able to work in a variety of arc welding processes. On the other hand, in some cases, it may be attracted by spurious edges. (Chapter 4, Section 4.6, published in EUVIP2011. (Liu et al., 2011c)).
5. We have also proposed the third active contour which is based on unifying both boundary-based and region-based information. We propose to use feature selective AdaBoost to select proper features to represent the pixels in the boundary and learn, by means of supervised machine learning techniques, a statistical model which can separate the true weld pool boundary from spurious edges. We then precede by combing this

boundary-based model and the region-based model as proposed in approach 2 and incorporating these two models into the framework of active contours. The new approach has been named as Feature Selective AdaBoosting Geodesic Active Region (FSA-GAR). The advantage is that its initialization is rather easy. The contour can start evolving from almost anywhere. It can tolerate intensive arc light and nevertheless capture the true weld pool boundary. (Chapter 4, Section 4.7).

6. We have proposed a new tangential diffusion term for stabilizing parametric active contours. To tackle the problem of numerical instability often occurring during evolution of parametric active contours, we investigate the previous work in literatures and proposed a new tangential diffusion term. We have theoretically proved that this term is bounded and therefore numerically stable. We have also experimentally demonstrated the superiority of this new term. (Chapter 5)
7. We have demonstrated the usability of the proposed vision system by feeding the visual information into the control system. We have investigated different visual servoing schemes for seam tracking and designed a proper one. The welding results verified the feasibility of the proposed vision system. (Chapter 6)

7.2 Future Work

In this section, we outline possible directions for extending the present work.

- **Improvement of Weld Pool Boundary Tracking Algorithms.** Deformable models have the advantage of taking prior knowledge into account. For instance in approach 3, the intensity prior and boundary classification prior have been incorporated into the weld pool tracking. Introducing more priors such as shape prior based on statistical shape models may increase the performance of the tracking algorithms.
- **Weld Pool Shape Analysis.** To reveal the information of the welding process contained by a series of weld pool shapes, we will study the techniques on analyzing the weld pool shapes. Existing researches on shapes analysis in the shape analysis community provide a variety of techniques for capturing the essential difference or variance of the shapes, for instance principle component analysis (PCA), independent component analysis (ICA) and manifold learning techniques (Stegmann and Gomez, 2002; Small, 1996). We shall investigate the possibilities of extracting

knowledge from a series of weld pool shapes using shape analysis techniques. It is also possible to employ other machine learning techniques such as hidden markov models (HMM) to detect unusual events during welding and/or generate control commands (Bishop, 2007; Jager et al., 2008; Jager and Hamprecht, 2009).

- **Closed-loop Control Based on the Proposed Vision System.** We will investigate more possible closed-loop control applications based on the proposed vision system such as the closed-loop seam tracker which can directly control the weld pool position and shape. We shall also study controlling the welding parameters based on the proposed vision system to achieve a good-quality welding.
- **Transfer to Other Welding Processes.** We will investigate applying the proposed techniques to other welding processes such as gas tungsten arc welding (GTAW) etc. In addition, we will also study the possibility of using the proposed visual sensing approach in laser and laser-hybrid welding.
- **Possible Applications Besides Control.** The primary purpose of developing the passive vision system presented in this dissertation has been to obtain visual information for the closed-loop control of arc welding processes. However, it may also be used in the cases such as online quality inspection or enhanced monitoring of the welding process.

APPENDIX A

Parametric Active Contours Driven by Region-based Statistical Models

In this appendix, we present how to minimize the region-based energy functional for parametric active contours using Calculus of Variation and Green's Theorem. The energy functional that is about to be minimized is as follows.

$$E(\mathcal{C}) = - \int_{\Omega_{\text{in}}} \log \frac{p_{\text{in}}(I(\mathbf{x}))}{p_{\text{out}}(I(\mathbf{x}))} d\mathbf{x} \quad (\text{A.1})$$

The gradient flow that minimizes the above energy functional is:

$$\log \frac{p_{\text{in}}(I(\mathbf{x}))}{p_{\text{out}}(I(\mathbf{x}))} \left\| \frac{d\mathbf{x}}{ds} \right\| \vec{\mathbf{n}} \quad (\text{A.2})$$

where $\vec{\mathcal{N}}$ is the outwards unit normal vector to C .

Proof: Let $\mathbf{f} = (P(x, y), Q(x, y))$ be a 2D vector field, Green's theorem shows that

$$\oint_C \mathbf{f} \cdot d\mathbf{r} = \oint_C \mathbf{f} \cdot \mathbf{T} ds = \iint_R \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right) dA \quad (\text{A.3})$$

where $\mathbf{T} = \frac{\mathbf{r}'(s)}{\|\mathbf{r}'(s)\|}$ is the unit tangent vector to C at $(x(s), y(s))$.

130 Parametric Active Contours Driven by Region-based Statistical Models

We choose a specific vector field \mathbf{f} subject to the following conditions:

$$P(x, y) = 0 \quad (\text{A.4})$$

$$\frac{\partial Q(x, y)}{\partial x} = \log \frac{p_{\text{in}}(I(x, y))}{p_{\text{out}}(I(x, y))} \quad (\text{A.5})$$

By using Green's theorem, we have

$$\begin{aligned} E'(\mathcal{C}) &= \iint_{\Omega_{\text{in}}} \log \frac{p_{\text{in}}(I(x, y))}{p_{\text{out}}(I(x, y))} dx dy \\ &= \oint_{\mathcal{C}} Q(x, y) dy \\ &= \int_0^1 Q(x, y) y'(s) ds \end{aligned} \quad (\text{A.6})$$

The *Gâteaux* derivative of $E'(\mathcal{C})$ is

$$\begin{aligned} \frac{dE(x + \epsilon\eta_1, y + \epsilon\eta_2)}{d\epsilon} \Big|_{\epsilon=0} &= \int_0^1 \frac{d(Q(x + \epsilon\eta_1, y + \epsilon\eta_2)(y' + \epsilon\eta_2'))}{d\epsilon} \Big|_{\epsilon=0} ds \quad (\text{A.7}) \\ &= \int_0^1 Q\eta_2' + Q_x\eta_1y' + Q_y\eta_2y' ds \\ &= \left(\int_0^1 (Q_x\eta_1y' + Q_y\eta_2y') ds + \int_0^1 Q\eta_2' ds \right) \\ &= \left(\int_0^1 (Q_x\eta_1y' + Q_y\eta_2y') ds + Q\eta_2 \Big|_{s=0}^{s=1} - \int_0^1 Q'\eta_2 ds \right) \\ &= \left(\int_0^1 (Q_x\eta_1y' + Q_y\eta_2y') ds - \int_0^1 (Q_x x'\eta_2 + Q_y y'\eta_2) ds \right) \\ &= \int_0^1 (Q_x y'\eta_1 - Q_x x'\eta_2) ds \\ &= \int_0^1 Q_x(y', -x') \cdot (\eta_1, \eta_2) ds \end{aligned}$$

To maximize the $E'(\mathcal{C})$, we set its *Gâteaux* derivative to be zero and have

$$Q_x(y', -x') = 0 \quad (\text{A.8})$$

Substituting Eq.A.5 to Eq.A.8, we have

$$\log \frac{p_{\text{in}}(I(x, y))}{p_{\text{out}}(I(x, y))} \parallel \frac{d\mathbf{x}}{ds} \parallel \vec{\mathbf{n}} = 0 \quad (\text{A.9})$$

where $\vec{\mathcal{N}}$ is the outwards unit normal vector to \mathcal{C} at $\mathbf{x} = (x, y)$. Hence, the *gradient* which maximizes $E'(\mathcal{C})$, i.e., minimizes $E(\mathcal{C})$, fastest is

$$\log \frac{p_{\text{in}}(I(x, y))}{p_{\text{out}}(I(x, y))} \left\| \frac{d\mathbf{x}}{ds} \right\| \vec{\mathbf{n}} \quad (\text{A.10})$$

APPENDIX B

Setting the Ground Truth: Manual Segmentation of Weld Pools

In this appendix, we shall discuss manual segmentation of weld pool boundaries which serve as ground truths in evaluating a weld pool boundary tracking algorithm quantitatively in terms of image analysis results.

In essence, the evaluation is based on comparison between shapes. In other words, we compare the extracted weld pool shape to the corresponding ground truth, i.e., the manually segmented shape. A common technique for shape comparison is so-called Procrustes analysis (Kendall, 1989). In general, shape comparison relies on two key elements. First, a certain adjustment, so called *shape alignment*, must be performed to eliminate the shape difference caused by such as the placement in space and the size of the shapes. Second, a certain metric must be defined properly to capture the shape difference, such as Procrustes distance. Mathematically, shape comparison can be expressed as follows.

$$\|\mathbf{X} - \mathbf{X}_G\|_\psi = \psi(\Gamma(\mathbf{X}), \Gamma(\mathbf{X}_G)) \quad (\text{B.1})$$

where \mathbf{X}_G stands for the manually segmented shape. $\psi(\cdot)$ or $\|\cdot\|_\psi$ denotes a certain metric. Γ denotes all allowed transformation of the shapes such as *translation, rotation, reflection and uniform scaling*.

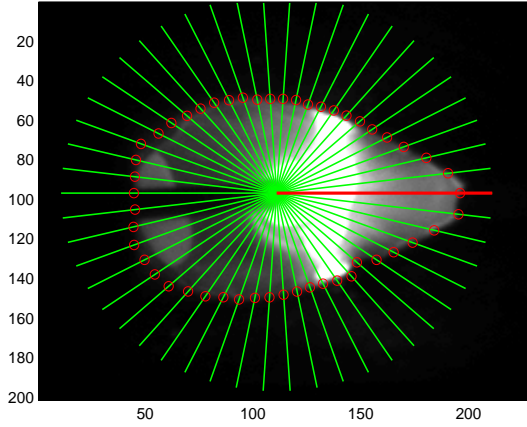


Figure B.1: An example of manually segmenting a weld pool boundary and representing all the points in a polar coordinate system (r, θ) . The resolution of the polar angle θ is 7.2° , i.e., 50 rays. The polar axis is indicated in a red line segment. The rays are marked in green. The landmark-like points which are extracted manually and describe the weld pool boundary are marked in red circles.

An alternative discussion on spline-based curve comparison can also be found in (Blake and Isard, 2000). Intensive researches have been made and are available in the literature. It is unnecessary to make a detailed review here. Instead, we shall concentrate on our problem and present our study regarding the shape comparison in our particular application.

Our goal here is, instead of capturing the shape variance as in shape analysis, to compare one shape to the ground truth. For each frame, a manual segmentation has been carried out and serves as a ground truth. Shape alignment is therefore not necessary here.

Now we shift our focus to design a proper distance metric. It is worth pointing out that the metric ψ can be selected very broadly. If we consider a curve as a function, any functional that satisfies the conditions required for being a distance is possible. However, if we, as in many shape analysis literatures, think of a shape as a set of special points, so called *landmarks*, it is natural to measure the shape difference on all the landmarks. The landmarks are special in the sense that they can characterize a shape somehow. For instance, if we describe the shape of a desk, possible landmarks are the corners. By linking all these corners, we will have a sketch of a desk. All the statistical measurement

can be performed on these landmarks. Using landmarks is very common in shape analysis of rigid objects (Small, 1996).

The weld pool in our application is however non-rigid fluid metal and its surface is more or less homogenous. It is difficult to describe a weld pool shape by regular landmark based methods, even though the weld pool boundary is indeed represented by a set of discrete points, i.e., a first order spline.

Therefore, we propose a measurement based on representing all the points in a polar coordinate system (r, θ) , as show in Fig.B.1. The pole is selected at the tip of the welding wire. It is convenient because the position of the welding wire tip in the image plane keeps unchange due to the fact the camera is mounted to the welding torch. The polar axis is determined as along the welding wire, see the red ray shown in Fig.B.1. The resolution of θ is determined as 7.2° , i.e., 50 rays in our application. All the control points extracted by an automatic algorithm or manually segmentation will be projected onto the closest ray. We can measure the difference of the shapes by measuring difference of the corresponding radial coordinates, i.e., $(\Delta r_1, \Delta r_2, \dots, \Delta r_N)$ where N is the amount of rays. This permits us to generate statistics such as mean and variance to compare two shapes quantitatively. For the evaluation of an algorithm, the *accuracy* or *error* can be defined as

$$\text{error} \equiv \begin{bmatrix} \Delta r_{11} & \Delta r_{12} & \dots & \Delta r_{1N} \\ \Delta r_{21} & \Delta r_{22} & \dots & \Delta r_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta r_{T1} & \Delta r_{T2} & \dots & \Delta r_{TN} \end{bmatrix} \quad (\text{B.2})$$

where T denotes the number of frames. N denotes the number of control points. A box plot computed in each row indicates how errors are distributed in each frame, i.e., the spatial error distribution. A box plot computed in each column indicates how errors are distributed at each control point, i.e., the temporal error distribution. The spatial-temporal error distributions are used in evaluating a welding pool boundary tracking approach in this dissertation.

Fig.B.1 gives an example of manually segmenting a weld pool boundary and representing all the points in a polar coordinate system. The polar axis is indicated in a red line segment. The rays are marked in green. The landmark-like points are marked in red circles. It is worth noting that shape analysis is usually performed in Cartesian coordinate systems. If necessary, we can transform all the points back to a Cartesian coordinate system.

APPENDIX C

On Parametrization

In this appendix, we shall discuss how the parametrization of a contour is characterized. For an arbitrary parametrization p , the relation between p and the natural parametrization, i.e., the *arc length* s is given as follows.

$$s = \int_{p_0}^{p_1} \left\| \frac{\partial \mathcal{C}}{\partial p} \right\| dp \quad (\text{C.1})$$

or

$$ds = \left\| \frac{\partial \mathcal{C}}{\partial p} \right\| dp \quad (\text{C.2})$$

Assume that the energy functional takes the form of

$$E(p) = \int_{\mathcal{C}} f(\mathcal{C}(p)) dp \quad (\text{C.3})$$

and

$$E(s) = \int_{\mathcal{C}} f(\mathcal{C}(s)) ds = \int_{p_0}^{p_1} f(\mathcal{C}(p)) \left\| \frac{\partial \mathcal{C}}{\partial p} \right\| dp \quad (\text{C.4})$$

It can be seen that the change of the energy functional from $E(p)$ to $E(s)$ is characterized by $\left\| \frac{\partial \mathcal{C}}{\partial p} \right\|$. In this sense, one can see that the parametrization is characterized by $g(p) = \left\| \frac{\partial \mathcal{C}}{\partial p} \right\|$. It is worth noting that if \mathcal{C} is considered as

the trajectory of a moving particle, $g(p, t)$ is in fact the velocity of the particle traveling along the curve.

For general energy functionals, the parametrization of a curve \mathcal{C} is characterized by

$$\frac{\partial \mathcal{C}(p)}{\partial p}, \frac{\partial^2 \mathcal{C}(p)}{\partial p^2}, \dots, \frac{\partial^n \mathcal{C}(p)}{\partial p^n} \quad (\text{C.5})$$

where p denotes the parametrization.

We shall choose a certain parametrization which can benefit the curve evolution. One common option is the *natural parametrization* s where the curve is parameterized by arc length, or the parametrization that is proportional to s , i.e.,

$$g(p) = \left\| \frac{\partial \mathcal{C}}{\partial p} \right\| = \zeta \quad (\text{C.6})$$

where ζ is a constant. $\zeta = 1$ leads to the *natural parametrization* s . In practice, the curve is often discretized into a set of control points. These kinds of parametrizations indicate that the control points of the curve are uniformly distributed. The advantage is that it can simplify the calculation of some geometric quantities such as the tangent vector, the normal vector and the curvature, and increase the stability of the curve evolution.

APPENDIX D

Textual Features

The textual features that are used as the pixel descriptor in the Feature Selective AdaBoosting Geodesic Active Region are as follows (Uppuluri, 2008; Soh and Tsatsoulis, 1999; Haralick et al., 1973).

1. Autocorrelation
2. Contrast
3. Correlation
4. Correlation
5. Cluster Prominence
6. Cluster Shade
7. Dissimilarity
8. Energy
9. Entropy
10. Homogeneity
11. Homogeneity

12. Maximum probability
13. Sum of squares
14. Sum average
15. Sum variance
16. Sum entropy
17. Difference variance
18. Difference entropy
19. Information measure of correlation
20. Inverse difference
21. Inverse difference normalized
22. Inverse difference moment normalized

Bibliography

- Abdullah, B., Smith, J., Lucas, W., Lucas, J., and Houghton, M. (2007). A low-cost vision system for real-time monitoring of welding applications. In *14th Int. Conf. on the Joining of Materials and 5th Int. Conf. on Education in Welding*, Helsingor, Denmark.
- Abdullah, B., Smith, J., Lucas, W., Lucas, J., and Malek, F. (2008). Monitoring of tig welding using laser and diode illumination sources: A comparison study. In *Electronic Design, 2008. ICED 2008. International Conference on*, pages 1 –4.
- Adalsteinsson, D. and Sethian, J. A. (1995). A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269 – 277.
- Alexandrov, O. (2004). http://en.wikipedia.org/wiki/Level_set_method.
- Asmar, N. H. (2004). *Partial Differential Equations with Fourier Series and Boundary Value Problems*. Prentice Hall, 2nd edition.
- Bae, K. Y., Lee, T. H., and Ahn, K. C. (2002). An optical sensing system for seam tracking and weld pool control in gas metal arc welding of steel pipe. *Journal of Materials Processing Technology*, 120(1-3):458 – 465.
- Balfour, C., Smith, J., and Al-SHAMMA'A, A. (2006). A novel edge feature correlation algorithm for real-time computer vision-based molten weld pool measurements. *Welding Journal*, 85(1):1.s–8.s.
- Bishop, C. M. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1st ed. 2006. corr. 2nd printing edition.

- Blake, A. and Isard, M. (2000). *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Springer, 1 edition.
- Brzakovic, D. and Khani, D. (1991). Weld pool edge detection for automated control of welding. *Robotics and Automation, IEEE Transactions on*, 7(3):397–343.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698.
- Caselles, V., Catte, F., Coll, T., and Dibos, F. (1993). A geometric model for active contours in image processing. *Numerische Mathematik*, 66:1–31. 10.1007/BF01385685.
- Chakraborty, A., Staib, L., and Duncan, J. (1996). Deformable boundary finding in medical images by integrating gradient and region information. *Medical Imaging, IEEE Transactions on*, 15(6):859–870.
- Cham, T.-J. and Cipolla, R. (1999). Automated b-spline curve representation incorporating mdl and error-minimizing control point insertion strategies. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(1):49–53.
- Chan, T. and Vese, L. (2001). Active contours without edges. *Image Processing, IEEE Transactions on*, 10(2):266–277.
- Chen, S. and Radke, R. (2009). Level set segmentation with both shape and intensity priors. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 763–770.
- Christensen, K. H. (2003). *Arc Welding Automation Process Modelling & Control Based on Neural Networks*. PhD thesis, Technical University of Denmark.
- Cohen, L. D. (1991). On active contour models and balloons. *CVGIP: Image Underst.*, 53(2):211–218.
- Delingette, H. and Montagnat, J. (2001). Shape and topology constraints on parametric active contours. *Computer Vision and Image Understanding*, 83:140–171.
- Felzenszwalb, P. F. and Huttenlocher, D. P. (2005). Pictorial structures for object recognition. *International Journal of Computer Vision*, 61:55–79.
- Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395.

- Fridenfalk, M. and Bolmsjo, G. (2003). Design and validation of a universal 6d seam tracking system in robotic welding based on laser scanning. *Industrial Robot: An International Journal*, 30(5):437 – 448.
- Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, 3(6):610 – 621.
- Hastie, T. and Tibshirani, R. (1998). Classification by pairwise coupling. In *Proceedings of the 1997 conference on Advances in neural information processing systems 10*, NIPS '97, pages 507–513, Cambridge, MA, USA. MIT Press.
- Hough, P. V. C. (1959). Machine analysis of bubble chamber pictures. In *International Conference on High Energy Accelerators and Instrumentation*.
- Houghton, M., Lucas, J., Lucas, W., and Smith, J. S. (2011). A pulsed laser camera system for weld pool observations. In *The 16th International Conference on the Joining of Materials JOM16 & the 7th International Conference on Education in Welding ICEW-7*, Tisvildeleje, Denmark.
- Hutchinson, S., Hager, G., and Corke, P. (1996). A tutorial on visual servo control. *Robotics and Automation, IEEE Transactions on*, 12(5):651 – 670.
- Illingworth, J. and Kittler, J. (1988). A survey of the hough transform. *Computer Vision, Graphics, and Image Processing*, 44(1):87 – 116.
- Jager, M. and Hamprecht, F. (2009). Principal component imagery for the quality monitoring of dynamic laser welding processes. *Industrial Electronics, IEEE Transactions on*, 56(4):1307 – 1313.
- Jager, M., Knoll, C., and Hamprecht, F. (2008). Weakly supervised learning of a classifier for unusual event detection. *Image Processing, IEEE Transactions on*, 17(9):1700 – 1708.
- Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331.
- Kendall, D. G. (1989). A survey of the statistical theory of shape. *Statistical Science*, 4(2):87–99.
- Kim, J. S., Son, Y. T., Cho, H. S., and Koh, K. I. (1995). A robust method for vision-based seam tracking in robotic arc welding. In *Intelligent Control, 1995., Proceedings of the 1995 IEEE International Symposium on*, pages 363 – 368.
- Kimia, B. B., Tannenbaum, A., and Zucker, S. W. (1992). On the evolution of curves via a function of curvature. i. the classical case. *Journal of Mathematical Analysis and Applications*, 163(2):438 – 458.

- Lefohn, A. E., Kniss, J. M., Hansen, C. D., and Whitaker, R. T. (2005). A streaming narrow-band algorithm: interactive computation and visualization of level sets. In *ACM SIGGRAPH 2005 Courses*, SIGGRAPH '05, New York, NY, USA. ACM.
- Lindeberg, T. (1998). Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30:79–116.
- Liu, J., Fan, Z., Olsen, S., Christensen, K., and Kristensen, J. (2011a). A vision system without external illuminations for seam tracking and weld pool sensing in arc welding. In *The 16th International Conference on the Joining of Materials JOM16 & the 7th International Conference on Education in Welding ICEW-7*, Tisvildeleje, Denmark.
- Liu, J., Fan, Z., Olsen, S., Christensen, K., and Kristensen, J. (2011b). Weld pool visual sensing in arc welding. In *The seventh annual IEEE Conference on Automation Science and Engineering*, Trieste, Italy.
- Liu, J., Fan, Z., Olsen, S., Christensen, K., and Kristensen, J. (2011c). Weld pool visual tracking using active contours driven by bayesian classifiers. In *The third European Workshop on Visual Information Processing (EUVIP 2011), Student Poster Sessions*, Paris, France.
- Ma, X. and Zhang, Y. (2009). Reflection of illumination laser from gas metal arc weld pool surface. *Measurement Science & Technology*, 20.
- Ma, X. and Zhang, Y. (2011). Gas metal arc weld pool surface imaging: Modeling and processing. *Welding Journal*, 90:85s – 94s.
- Maji, S. and Malik, J. (2009). Object detection using a max-margin hough transform. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1038 –1045.
- Malis, E. (2002). Survey of vision-based robot control. In *ENSIETA European Naval Ship Design Short Course*.
- Malis, E., Chaumette, F., and Boudet, S. (1999). 2 1/2 d visual servoing. *Robotics and Automation, IEEE Transactions on*, 15:238–250.
- Malladi, R., Sethian, J., and Vemuri, B. (1995). Shape modeling with front propagation: a level set approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(2):158 –175.
- Menet, S., Saint-Marc, P., and Medioni, G. (1990). Active contour models: overview, implementation and applications. In *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on*, pages 194 –199.

- Mikula, K. and sevcovic, D. (2004). Computational and qualitative aspects of evolution of curves driven by curvature and external force. *Computing and Visualization in Science*, 6:211–225.
- Olivier, J., Bon, R., Rousselle, J.-J., and Cardot, H. (2008a). Active contours driven by supervised binary classifiers for texture segmentation. In *Advances in Visual Computing*, volume 5358 of *Lecture Notes in Computer Science*, pages 288–297. Springer Berlin / Heidelberg.
- Olivier, J., Bone, R., Rousselle, J.-J., and Cardot, H. (2008b). Active contours driven by supervised binary classifiers for texture segmentation. In *Advances in Visual Computing*, volume 5358 of *Lecture Notes in Computer Science*, pages 288–297. Springer Berlin / Heidelberg.
- Osher, S. J. and Fedkiw, R. P. (2002). *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 1 edition.
- Otsu, N. (1979). A threshold selection method from gray level histograms. *IEEE Trans. Systems, Man and Cybernetics*, 9:62–66. minimize inter class variance.
- Paragios, N. and Deriche, R. (2002a). Geodesic active regions: A new framework to deal with frame partition problems in computer vision. *Journal of Visual Communication and Image Representation*, 13(1-2):249 – 268.
- Paragios, N. and Deriche, R. (2002b). Geodesic active regions and level set methods for supervised texture segmentation. *Int. J. Comput. Vision*, 46:223–247.
- Paragios, N. and Deriche, R. (2005). Geodesic active regions and level set methods for motion estimation and tracking. *Computer Vision and Image Understanding*, 97(3):259 – 282.
- Roberts, M., Packer, J., Sousa, M. C., and Mitchell, J. R. (2010). A work-efficient gpu algorithm for level set segmentation. In *Proceedings of the Conference on High Performance Graphics*, HPG '10, pages 123–132, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.
- Shen, H., Wu, J., Lin, T., and Chen, S. (2008). Arc welding robot system with seam tracking and weld pool control based on passive vision. *The International Journal of Advanced Manufacturing Technology*, 39:669–678.
- Shi, F., Huang, X., Duan, Y., and Chen, S. (2010). Part-based model for visual detection and localization of gas tungsten arc weld pool. *The International Journal of Advanced Manufacturing Technology*, 47:1097–1104.
- Shi, F., Lin, T., and Chen, S. B. (2009). Efficient weld seam detection for robotic welding based on local image processing. *Industrial Robot: An International Journal*, 36:277–283.

- Sicard, P. and Levine, M. (1989). Joint recognition and tracking for robotic arc welding. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(4):714–728.
- Silapachote, P., Karuppiiah, D. R., and Hanson, A. R. (2005). Feature selection using adaboost for face expression recognition.
- Small, C. G. (1996). *The Statistical Theory of Shape*. Springer.
- Soh, L.-K. and Tsatsoulis, C. (1999). Texture analysis of sar sea ice imagery using gray level co-occurrence matrices. *Geoscience and Remote Sensing, IEEE Transactions on*, 37(2):780–795.
- Song, H. and Zhang, Y. (2007a). Image processing for measurement of three-dimensional gta weld pool surface. *Welding Journal*, 86:323s–330s.
- Song, H. and Zhang, Y. (2007b). Three-dimensional reconstruction of specular surface for gas tungsten arc weld pool. *Measurement Science & Technology*, 18:3751–3767.
- Song, H. and Zhang, Y. (2008). Measurement and analysis of three-dimensional specular gas tungsten arc weld pool surface. *Welding Journal*, 87:85s–95s.
- Song, H. and Zhang, Y. (2009). Error analysis of a three-dimensional gta weld pool surface measurement system. *Welding Journal*, 88:141s–148s.
- Srikrishnan, V. and Chaudhuri, S. (2009). Stabilization of parametric active contours using a tangential redistribution term. *Image Processing, IEEE Transactions on*, 18(8):1859–1872.
- Stegmann, M. B. and Gomez, D. D. (2002). A brief introduction to statistical shape analysis. Images, annotations and data reports are placed in the enclosed zip-file.
- Uppuluri, A. (2008). <http://www.mathworks.com/matlabcentral/fileexchange/22187-g lcm-texture-features>.
- Wang, J. J., Lin, T., and Chen, S. B. (2005). Obtaining weld pool vision information during aluminium alloy tig welding. *The International Journal of Advanced Manufacturing Technology*, 26:219–227.
- Wang, Z., Ma, X., and Zhang, Y. (2011). Simultaneous imaging and measurement of pool surface and metal transfer. *Welding Journal*, 90:121s–128s.
- Wang, Z., Zhang, Y., and Wu, L. (2010). Measurement and estimation of weld pool surface depth and weld penetration in pulsed gas metal arc welding. *Welding Journal*, 89(6):117s–126s.

- Whitney, D. E. (1972). The mathematics of coordinated control of prosthetic arms and manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 94(4):303–309.
- Xu, C. and Prince, J. (1998). Snakes, shapes, and gradient vector flow. *Image Processing, IEEE Transactions on*, 7(3):359 –369.
- Xu, C. and Prince, J. L. (2000). Medical image segmentation using deformable models. pages 129–174.
- Xu, P., Xu, G., Tang, X., and Yao, S. (2008). A visual seam tracking system for robotic arc welding. *The International Journal of Advanced Manufacturing Technology*, 37:70–75. 10.1007/s00170-007-0939-6.
- Zhang, G., Yan, Z., and Wu, L. (2006). Visual sensing of weld pool in variable polarity tig welding of aluminium alloy. *Transactions of Nonferrous Metals Society of China*, 16(3):522 – 526.
- Zhang, Q., Yeo, T. S., Tan, H. S., and Luo, Y. (2008). Imaging of a moving target with rotating parts based on the hough transform. *Geoscience and Remote Sensing, IEEE Transactions on*, 46(1):291 –299.
- Zhao, D., Chen, S., Wu, L., and Chen, Q. (2003). Extraction of three-dimensional parameters for weld pool surface in pulsed gtaw with wire filler. *J. Manuf. Sci. Eng.*, 125(3):493–503.
- Zhu, S. C. and Yuille, A. (1996). Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18:884–900.