

AN ABSTRACT OF THE THESIS OF

James S. Yih for the degree of Master of Science in Mechanical Engineering presented on November 29, 2011.

Title: The Geometric Characterization and Thermal Performance of a Microchannel Heat Exchanger for Diesel Engine Waste Heat Recovery

Abstract approved:

Richard B. Peterson

Rising energy demands and the continual push to find more energy efficient technologies have been the impetus for the investigation of waste heat recovery techniques. Diesel engine exhaust heat utilization has the potential to significantly reduce the consumption of fossil fuels and reduce the release of greenhouse gases, because diesel engines are ubiquitous in industry and transportation. The exhaust energy can be used to provide refrigeration by implementing an organic Rankine cycle coupled with a vapor-compression cycle. A critical component in this system, and in any waste heat recovery system, is the heat exchanger that extracts the heat from the exhaust.

In this study, a cross-flow microchannel heat exchanger was geometrically examined and thermally tested under laboratory conditions. The heat exchanger, referred to as the Heat Recovery Unit (HRU), was designed to transfer diesel exhaust energy to a heat transfer oil. Two methods were developed to measure the geometry of the microchannels. The first was based on image processing of microscope photographs, and the second involved an analysis of profilometer measurements. Both methods revealed that the exhaust channels (air channels) were, on average, smaller in cross-sectional area by 11% when compared to the design. The

cross-sectional area of the oil channels were 8% smaller than their design. The hydraulic diameters for both channel geometries were close to their design.

Hot air was used to simulate diesel engine exhaust. Thermal testing of the heat exchanger included measurements of heat transfer, effectiveness, air pressure drop, and oil pressure drop. The experimental results for the heat transfer and effectiveness agreed well with the model predictions. However, the measured air pressure drop and oil pressure drop were significantly higher than the model. The discrepancy was attributed to the model's ideal representation of the channel areas. Additionally, since the model did not account for the complex flow path of the oil stream, the measured oil pressure drop was much higher than the predicted pressure drop. The highest duty of the Heat Recovery Unit observed during the experimental tests was 12.3 kW and the highest effectiveness was 97.8%.

To examine the flow distribution through the air channels, velocity measurements were collected at the outlet of the Heat Recovery Unit using a hot film anemometer. For unheated air flow, the profile measurements indicated that there was flow maldistribution. A temperature profile was measured and analyzed for a thermally loaded condition.

©Copyright by James S. Yih
November 29, 2011
All Rights Reserved

The Geometric Characterization and Thermal Performance of a Microchannel Heat Exchanger
for Diesel Engine Waste Heat Recovery

by

James S. Yih

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented November 29, 2011
Commencement June 2012

Master of Science thesis of James S. Yih presented on November 29, 2011.

APPROVED:

Major Professor, representing Mechanical Engineering

Head of the School of Mechanical, Industrial, and Manufacturing Engineering

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

James S. Yih, Author

ACKNOWLEDGEMENTS

I express my sincere gratitude to Dr. Richard Peterson who provided me with the opportunity to work with him and his research group at the Microproducts Breakthrough Institute (MBI). Thank you for giving me the freedom, funding, and advice I needed to explore my research topic. I also want to thank you for nominating me for the scholarship that I received from the Mechanical Engineering department.

I have been very fortunate to work alongside Dr. Hailei Wang. Dr. Wang was not only very insightful about our research, but he always had time to answer my random questions. Dr. Wang, it is always a pleasure working with you. And thank you for politely telling me that I had been saying my name wrong my entire life.

I am indebted to Dr. Jair Lizarazo, Don Higgins, and Dr. Shankar Krishnan of Pacific Northwest National Laboratory (PNNL) for all of their guidance and inspiration throughout my stay in their laboratory. I have learned so much from you. Without them, I would not have been able to construct my test facility to their level of excellence.

It is also my pleasure to thank Dr. Jack Rundel, formerly of MBI and OSU, for always finding the time in his busy schedule to answer my questions or help me find something. How he ever got anything done with the constant bombardment of questions from staff and students alike, I will never know.

I would like to acknowledge my fellow graduate colleagues Ryan Seward and Chris Ward. To Ryan, who helped with my experimental setup, and Chris, who designed and modeled the Heat Recovery Unit, thank you.

From my friends and colleagues Kevin Harada, Robbie Ingram-Goble, and Erik Miller, I have learned much about research, technology, and life in general. Thank you for giving me the motivation for, and the times of respite from, my journey as a graduate student.

The thermal and fluid science professors at OSU were the ones who greatly shaped my educational experience as an undergraduate and as a graduate student. Words cannot express how grateful I am to Dr. Nancy Squires. She has always greeted me with a warm welcome and a sincere smile. Not only did Dr. Squires provide excellent help for the many classes I took under her instruction, but she also allowed me to work as teaching assistant for her classes during my undergraduate and graduate careers. Being a MATLAB teaching assistant for Dr. Squires was one of the most rewarding experiences I have ever had. Thank you, Dr. Squires, for all of your guidance and encouragement. I would also like to thank Dr. Vinod Narayanan, Dr. Deborah Pence, and Dr. James (Jim) Liburdy for helping me become an engineer, and for making me realize how little I truly know about thermal and fluid sciences. I also thank Dr. Nathan Gibson, of the math department, for serving as my Graduate Council Representative.

Of course, I would not be here without the love and support from my family. I particularly want to thank my brother for making me food for all the late nights I spent in the lab. It is with great appreciation that I present to them my thesis, not only as a symbol of my accomplishments, but theirs as well.

To all of my friends and family, I greatly appreciate your support.

Thank you all!

TABLE OF CONTENTS

	<u>Page</u>
1 Introduction	1
2 Review of Technology	3
2.1 Rankine and Vapor-Compression Cycles	3
2.2 Waste Heat Recovery	8
2.3 Heat Exchangers for Waste Heat Recovery	10
2.4 Microchannel Devices	14
2.4.1 Description of Microchannel Devices	14
2.4.2 Fabrication of Microchannel Devices	16
2.5 Microchannel Profiling and Imaging	19
3 Heat Exchanger Design	21
3.1 Waste Heat Recovery System	21
3.2 Heat Recovery Unit Design	23
3.3 Thermal Modeling	26
3.4 Fabrication Results	29
4 Scope and Objectives	32
5 Heat Exchanger Testing	33
5.1 Microscope Image Analysis	33
5.1.1 Microscope System	33
5.1.2 Microscope Calibration	34
5.1.3 Measurement Process	35
5.2 Shim Profile Analysis	36
5.2.1 ZeScope Optical Profilometer	36
5.2.2 Measurement Process	37
5.3 Thermal Testing Facility	39
5.3.1 Test Stand and Components	39
5.3.2 Thermal Test Matrix	43
5.3.3 Traverse Measurements	44
5.3.4 Equipment Calibration	48
5.3.4.1 Hot Wire Anemometer Calibration	48
5.3.4.2 Flow Meter Calibration	49
6 Data Analysis	51
6.1 Data Reduction and Calculations	51

TABLE OF CONTENTS (Continued)

	<u>Page</u>
6.1.1 Microscope Image Analysis.....	51
6.1.2 Shim Profile Analysis	56
6.1.2.1 Channel Geometry Calculator	58
6.1.2.2 Multi-scan Leveler and Stitcher.....	65
6.1.3 Thermal Performance	73
6.1.4 Velocity and Temperature Profiles	75
6.2 Uncertainty Analysis	81
6.2.1 Uncertainties in Microscope Image Processing.....	81
6.2.2 Uncertainties in Profilometer Processing.....	84
6.2.3 Uncertainties in Thermal Performance	86
6.2.4 Uncertainties in Velocity and Temperature Profiles.....	89
7 Results and Discussion	91
7.1 Microscope Image Analysis	91
7.2 Shim Profile Analysis	94
7.3 Heat Exchanger Performance Tests.....	97
7.3.1 Thermal Tests	97
7.3.1.1 Duty	97
7.3.1.2 Effectiveness.....	101
7.3.1.3 Air Pressure Drop	103
7.3.1.4 Oil Pressure Drop	105
7.3.1.5 Oil Inlet Temperature Comparison	107
7.3.2 Hot Wire Tests.....	110
7.3.2.1 Horizontal and Vertical Scans	110
7.3.2.2 Window Scans	116
7.3.2.3 Full Field Maximum Velocity Interpolation.....	118
7.3.3 Thermal Profile.....	119
8 Conclusion and Future Work.....	122
References	125
Appendices	130

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
2.1. Diagram of a basic Rankine cycle.....	4
2.2. Diagram of a basic vapor-compression refrigeration cycle.....	5
2.3. Types of working fluids	7
2.4. Heat pipe operation	11
2.5. Shell-and-tube heat exchanger	12
2.6. Single-pass, cross-flow plate-fin heat exchanger	13
2.7. Photochemical etching process (with positive or negative photoresist).....	18
2.8. Photochemical etching profile artifacts	19
3.1. Waste heat recovery system	22
3.2. Two-pass, cross-counter flow plate-fin heat exchanger	23
3.3. SolidWorks model of the Heat Recovery Unit.....	23
3.4. Exhaust shim	24
3.5. Oil shim.....	25
3.6. Thermal model diagram	27
3.7. Model heat transfer and pressure drop regions for both shim geometries.....	28
3.8. A photograph of the Heat Recovery Unit.....	30
3.9. A photograph of the exhaust channels	30
3.10. Exhaust shim photograph.....	31
3.11. Oil shim photograph.....	31
5.1. A photograph of the microscope testing facility	34
5.2. Example calibration image for the x-direction at 10x.....	35
5.3. Microscope reference image at 10x.....	36
5.4. Profilometer scan regions.....	37
5.5. ZeScope screenshot of oil shim channels and their profiles.....	38
5.6. Heat Recovery Unit thermal test bench schematic.....	39

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
5.7. Manifold photograph.....	40
5.8. Thermal testing facility	41
5.9. Scanning sequence styles	44
5.10. Coordinate system for the traverse measurements.....	45
5.11. Air channel numbering and spacing.....	45
5.12. Hot wire measurement regions: vertical and horizontal regions shown on the left, window regions shown on the right	46
5.13. Velocity in the z-direction.....	48
5.14. Hot wire calibration system.....	49
5.15. Oil flow meter calibration system	50
6.1. Example microscope image of the air channels	52
6.2. Convert image to grayscale.....	52
6.3. Adjust contrast	52
6.4. Convert image to black and white (binary)	53
6.5. Dilate the image	53
6.6. Channel edge detection	53
6.7. Incomplete channels removed.....	53
6.8. Small particles removed	54
6.9. Filled and smoothened channels	54
6.10. Line and arc masks on a channel.....	55
6.11. Masks incorporated and channels cleaned	55
6.12. Example result of the Microscope Image Processing	55
6.13. Profilometer processing diagram.....	57
6.14. Example profile data	58
6.15. Profile data and the first derivative	59

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
6.16. Identify derivatives near zero	59
6.17. Tops (cyan) and bottoms (green) separated by the horizontal line in the x-y data.....	60
6.18. Change in x for top points	61
6.19. Edges of the shim tops marked in red	61
6.20. Line fit on shim top	62
6.21. Left and right ends of a channel top	62
6.22. Channel tops	63
6.23. Integration area of the channel top line	63
6.24. Integration areas of the channel top line and the channel surface	64
6.25. Channel area	64
6.26. Example data to level	66
6.27. Channel lines and shim top lines drawn using the methods previously developed.....	66
6.28. Shim top lines extended for the left and right ends	67
6.29. Level line at maximum y value	67
6.30. Leveled data	68
6.31. Two leveled scans to stitch.....	68
6.32. Trimmed scans	69
6.33. Both scans plotted together	69
6.34. Scan 2 shifted right.....	70
6.35. Scan 2 leveled with Scan 1	70
6.36. Overlap region.....	71
6.37. Left and right ends of the overlap region with end points circled and points for merging connected by a line.....	71
6.38. Calculated merge points (magenta).....	72
6.39. Merge result	72

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
6.40. Stitch complete.....	73
6.41. Averaging scheme for the velocity and temperature profiles.....	76
6.42. 3-D plot of the horizontal center velocity scan (left) and a plot of the collapsed data (right).....	77
6.43. First derivative of collapsed data.....	78
6.44. Maximum channel velocities for the horizontal center scan.....	78
6.45. Intersecting points of the horizontal and vertical velocity scans.....	80
6.46. An illustration of the points used for calculating the shift lines with respect to a vertical scan.....	81
6.47. A comparison between the results of the automated outlining (left) and the manual outlining (right) methods.....	83
6.48. A comparison between the average profiles of a straight channel and an angled channel	84
6.49. The unaccounted area in a shim profile.....	85
7.1. An example of one of the best results from the Microscope Image Analysis.....	91
7.2. An example of one of the worst outlining results from the Microscope Image Analysis.....	92
7.3. Air channel areas from the Microscope Image Analysis.....	93
7.4. Air channel profile	94
7.5. Oil channel profile.....	94
7.6. Air shim cross-sectional channel area results.....	95
7.7. Oil shim cross-sectional channel area results.....	95
7.8. Duty for 40 g/s warm oil flow	97
7.9. Duty for 47 g/s warm oil flow	98
7.10. Duty for 55 g/s warm oil flow	98
7.11. A comparison between the air energy and oil energy (warm oil cases).....	99
7.12. Effectiveness for 40 g/s warm oil flow	101

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
7.13. Effectiveness for 47 g/s warm oil flow	102
7.14. Effectiveness for 55 g/s warm oil flow	102
7.15. Air pressure drop	103
7.16. Oil pressure drop	106
7.17. Duty for 47 g/s oil at inlet temperatures of 73 °C (warm) and 53 °C (cold)	108
7.18. Effectiveness for 47 g/s oil at inlet temperatures of 73 °C and 53 °C	108
7.19. Oil pressure drop comparison for oil at 47 g/s	109
7.20. 3-D plots of horizontal and vertical velocity scans	111
7.21. Side views of the horizontal and vertical velocity plots	112
7.22. Contour plots of the horizontal velocity scans	113
7.23. Contour plots of the vertical velocity scans	114
7.24. Velocity contour plot of the center window	117
7.25. Velocity contour plot of the bottom left window	117
7.26. Full field maximum velocity interpolation result	119
7.27. Full field temperature profile for conditions close to the design	120

LIST OF TABLES

<u>Table</u>	<u>Page</u>
3.1. Channel dimensions and information of the shims	26
3.2. Heat Recovery Unit design conditions and model results.....	28
5.1. Equipment information and specifications.....	41
5.2. Instrumentation information and specifications	42
5.3. Thermal test matrix	43
5.4. Scan resolutions.....	47
5.5. Equipment used to calibrate the signal conditioners	50
6.1. Area and perimeter results of the two outlining methods	83
6.2. Average relative uncertainty of measured quantities	88
6.3. Average relative uncertainty of calculated quantities	89
7.1. Statistical summary of the Microscope Image Processing results.....	93
7.2. Summary of Profilometer Processing results	96
7.3. Reynolds number and entry length for the air channels at the design flow conditions and measured geometry	104
7.4. Reynolds number and entry length for the oil channels at the design flow conditions and measured geometry	107
7.5. Test conditions for the full field temperature profile	119

LIST OF APPENDICES

<u>Appendix</u>	<u>Page</u>
Appendix A Additional Results	131
A.1 Additional Microscope Image Processing Results	131
A.2 Additional Shim Profile Analysis Results	132
A.3 Cold Oil Thermal Test Results	134
A.3.1 Duty	134
A.3.2 Effectiveness.....	135
A.3.3 Air Pressure Drop	137
A.3.4 Oil Pressure Drop	137
A.4 Additional Oil Inlet Temperature Comparisons	138
A.4.1 Duty Comparison.....	138
A.4.2 Effectiveness Comparison	139
A.4.3 Oil Pressure Drop Comparison	140
Appendix B Maximum Velocity Profiles.....	141
Appendix C MATLAB Code.....	142
C.1 Microscope Image Analysis	142
C.1.1 microscope_image_processing.m	142
C.1.2 create_line.m.....	148
C.1.3 create_arc.m.....	150
C.1.4 sort_regions.m	153
C.2 Shim Profile Analysis.....	155
C.2.1 profile_area_calculator.m	155
C.2.2 profile_multi_scan_level_and_stitch.m.....	162
C.3 Velocity Interpolation.....	174
C.3.1 Velocity_Profile_Interpolator.m.....	174
C.3.2 inpaint_nans.m.....	177
C.4 Common Function Files	178
C.4.1 Central_Difference.m	178
C.4.2 Average_Matrix.m.....	179

LIST OF APPENDIX FIGURES

Appendix A

<u>Figure</u>	<u>Page</u>
A.1. Air channel perimeters from the Microscope Image Processing.....	131
A.2. Air channel hydraulic diameters from the Microscope Image Processing.....	131
A.3. Air shim perimeters.....	132
A.4. Air shim hydraulic diameters.....	132
A.5. Oil shim perimeter results.....	133
A.6. Oil shim hydraulic diameter results.....	133
A.7. Duty for 40 g/s cold oil flow.....	134
A.8. Duty for 47 g/s cold oil flow.....	134
A.9. Duty for 55 g/s cold oil flow.....	135
A.10. Effectiveness for 40 g/s cold oil flow.....	135
A.11. Effectiveness for 47 g/s cold oil flow.....	136
A.12. Effectiveness for 55 g/s cold oil flow.....	136
A.13. Air pressure drop for cold oil.....	137
A.14. Oil pressure drop for cold oil.....	137
A.15. Duty comparison for 40 g/s oil.....	138
A.16. Duty comparison for 55 g/s oil.....	138
A.17. Effectiveness comparison 40 g/s oil.....	139
A.18. Effectiveness 55 g/s oil.....	139
A.19. Oil pressure drop comparison for 40 g/s oil flow.....	140
A.20. Oil pressure drop comparison for 55 g/s oil flow.....	140

Appendix B

<u>Figure</u>	<u>Page</u>
B.1. Extracted maximum channel velocities: horizontal scans on the left and vertical scans with shifts on the right.....	141

1 Introduction

Worldwide demand for energy continues to grow as the global population increases, economies expand, and the need increases for industrial products. Scientists and engineers constantly search for technologies that will help solve prominent energy issues. These include meeting high energy demands as supplies diminish, reducing the environmental impacts of fuel consumption, and reducing energy costs. One of the technologies where improvements could make a significant impact in these energy issues is energy recovery systems.

Energy utilization plays a key role in industry, where power production and providing heating and cooling are but two of the many processes found. One of the important elements in industrial energy systems is the internal combustion engine. Typically, internal combustion engines have efficiencies between 20–40%. That means, at best, 60% of the fuel energy is lost as waste heat and conversion inefficiencies. If this waste heat could be captured and utilized, it could significantly impact energy usage on a large scale. In the United States, the industrial and transportation sectors account for nearly 60% of the total energy consumption [1]. In 2010, the combined energy consumption of the two sectors was 57.7 trillion BTU [1]. Because internal combustion engines, and especially the diesel engine, are literally the driving force behind these sectors, even small improvements in engine efficiency could mean substantial energy savings. A system for diesel engine waste heat recovery could also reduce the release of greenhouse gases into the atmosphere.

The exhaust heat from a diesel engine is considered medium grade heat, with temperatures ranging between 300 to 500 °C. This level of energy has the potential to be converted into useful heat, electricity, or cooling. In the application concerning this research,

diesel engine exhaust is used to produce cooling. Cooling is accomplished by utilizing waste heat to power an Organic Rankine Cycle (ORC) that is coupled to a vapor-compression refrigeration cycle.

The focus of this study is to examine the performance of the heat exchanger that interfaces between the diesel engine and the organic Rankine cycle. The Heat Recovery Unit (HRU) is a cross-flow microchannel heat exchanger designed and fabricated for this application. In this thesis, the manufacturing accuracy and thermal performance of the HRU is determined.

Two techniques were developed for measuring the microchannel geometries of the Heat Recovery Unit. The first method used an image processing analysis based on pictures taken from a microscope. This method was capable of measuring microchannel areas on devices with visually accessible channels. The second technique utilized measurements from an optical profilometer to determine channel profiles from shims that contained the microchannel patterns. Results, limitations, and comparisons of these two methods are presented.

The Heat Recovery Unit was then tested under laboratory conditions using a hot air source to simulate diesel exhaust. The thermal performance was measured under various conditions to validate the theoretical design model. Velocity and temperature profiles were also measured at the exhaust exit plane for the examination of the flow distribution.

2 Review of Technology

Technologies that relate to the work of this study are presented, starting with a global perspective of how this work applies to energy systems and waste heat recovery. A brief review of Rankine and vapor-compression cycles is given, followed by a discussion of their applications in waste heat recovery. Next, several different types of heat exchangers are discussed in terms of their role in utilizing waste heat. Since the heat exchanger in this study takes advantage of microchannels, microchannel technology and fabrication methods are examined. Finally, previous experiments relating to microchannel profiling and imaging are discussed.

2.1 Rankine and Vapor-Compression Cycles

A Rankine cycle is a closed loop vapor power cycle. Its purpose is to convert thermal energy of a working fluid into work. One of the earliest notable applications of a Rankine cycle was the steam engine, which powered boats, railroad trains, and industrial equipment during the 18th and 19th centuries. Today, Rankine cycles are predominately used to generate electricity in coal-fired, nuclear, and solar-thermal power plants.

A schematic of a Rankine cycle is shown in Figure 2.1. There are four main components in a basic Rankine cycle: the pump, boiler, expander, and condenser. The working fluid enters the pump in a liquid state where it is pressurized and sent to the boiler. At the boiler, heat is added to vaporize the fluid. Energy from the high temperature, high pressure vapor is extracted by allowing the vapor to expand through the turbine (or expander) to lower temperatures and pressures. Once through the turbine, the working fluid passes through a condenser where it is cooled back to a liquid and sent to the pump, thus completing the cycle.

Cycle efficiencies can be improved by adding recuperative or regenerative heat exchangers for cycles involving certain working fluids, typically large molecular weight refrigerants. Also, multi-stage turbines can be used to extract more work from the expanding fluid [2] in large scale power plant applications.

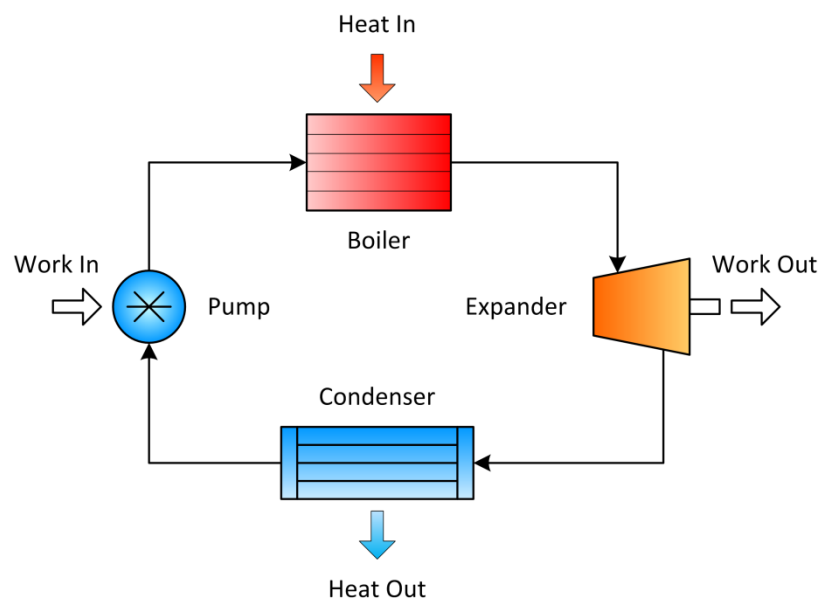


Figure 2.1. Diagram of a basic Rankine cycle

Steam-based Rankine cycles require a large amount of heat to operate because of the high latent heat of vaporization of water. Rankine cycles with organic working fluids can operate with lower heat inputs and are good candidates for utilizing low to moderate temperature waste heat [3], [4].

Vapor-compression cycles are used in refrigeration applications, such as food preservation, building and vehicle air conditioning, and industrial cooling processes. A vapor-compression cycle has components similar to a Rankine cycle, but the thermodynamic process of the working fluid is different. A diagram of a classic vapor-compression cycle is shown in

Figure 2.2. The primary components in a vapor-compression cycle are the compressor, condenser, expansion valve, and evaporator.

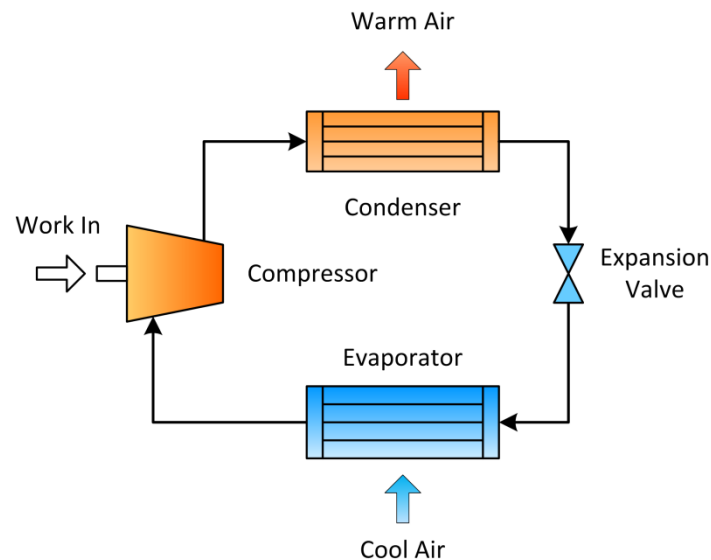


Figure 2.2. Diagram of a basic vapor-compression refrigeration cycle

Starting at the compressor, saturated vapor enters and becomes a superheated vapor as temperature and pressure are increased. Next, the superheated vapor loses heat in the condenser and, ideally, becomes a saturated liquid. After the condenser, the fluid is sent to an expansion valve where the temperature and pressure decrease in preparation for the evaporator. The expansion valve, also referred to as a thermal expansion valve, controls the amount of flow through the evaporator. The evaporator is where the cooling effect is felt. The surrounding area that is being cooled gives heat to the working fluid in the evaporator because the working fluid is at a lower temperature. As the fluid gains energy, it vaporizes at a constant temperature, which allows the cooling to continue. The cycle is complete once the vapor returns to the compressor.

Vapor-compression cycles rely on working fluids with a low boiling point and high latent heat of vaporization to extract heat from the environment. Some common vapor-compression working fluids are hydrofluorocarbons (HFC), such as Refrigerant 134a, and hydrocarbons, such as propane (C_3H_8) and methane (CH_4). Ammonia-based refrigeration cycles are also under investigation. These fluids are preferred over the chlorofluorocarbons, which have been historically used in refrigeration cycles, because they have a lower impact on ozone depletion [2].

Conventional macroscale Rankine cycles used in power plants are very large and immobile. In order for a combined organic Rankine cycle and vapor-compression cycle to be considered for waste heat applications in the transportation sector, the systems must be scaled for mobility. There are some challenges in directly scaling conventional Rankine cycle components, particularly the turbine. At lower temperatures and flow rates, turbine efficiencies become undesirable. One promising technology that addresses this issue is the scroll expander. Small scale (1–10 kW) scroll-based expanders and compressors have demonstrated reasonable efficiencies both individually and when integrated into a combined cycle. Wang et al. [5] and Harada [6] both demonstrated designs of scroll expanders that produced 1 kW of power, in the form of shaft work, and achieved isentropic efficiencies over 70%. When integrated in a combined organic Rankine cycle and vapor-compression cycle, the scroll expander by Wang et al. [7] reached an isentropic efficiency up to 84%. Lemort et al. [8] developed a model for their 1–2 kW scroll expander and Quoilin et al. [9] tested it with agreeable results.

Another challenge in the scalability of the Rankine cycle is the choice of working fluid. The selection of a working fluid is critical to the efficiency of a Rankine cycle, and is dominated by the temperatures of the operating environments. Numerous studies have been conducted to investigate the impact of various working fluids on the cycle performance. Common applications are geothermal energy utilization, solar energy collection, and low-grade waste heat recovery [4], [10–13]. Working fluids can be classified based on the slope of their saturation vapor curve on a T-s diagram, as shown in Figure 2.3. A wet fluid has a negative slope, an isentropic fluid has a nearly infinite slope (approximately vertical), and a dry fluid has a positive slope. The expansion behavior of these fluids is strongly dependent on their saturation curve. When a wet fluid expands isentropically from a superheated vapor, it may start to condense as it falls into the liquid-vapor region of the T-s diagram. When an isentropic fluid is expanded as a slightly superheated vapor, the fluid tends to stay near the saturated vapor line. A dry fluid stays a superheated vapor after an isentropic expansion in the superheated vapor region.

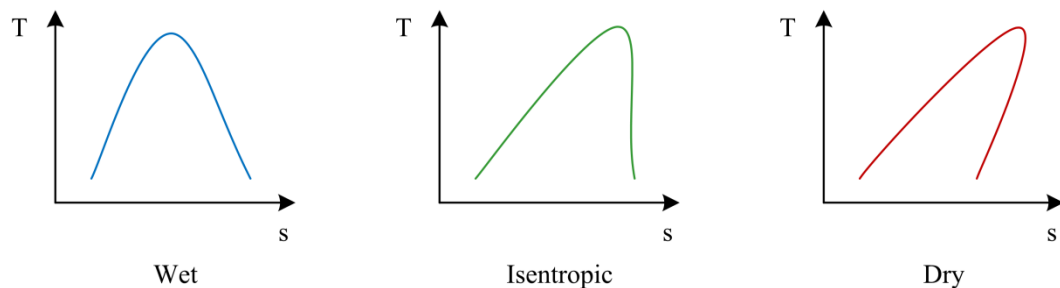


Figure 2.3. Types of working fluids

Wang et al. [13] simulated the performance of a variety of fluids and concluded that although some of the evaluated refrigerants showed better thermal performance, R245fa and R245ca are better practical choices because they are nontoxic and have a low environmental impact. For

organic Rankine cycles to be effective, the thermal performance and environmental impact must be balanced.

2.2 Waste Heat Recovery

In general, there are three levels of waste heat that are loosely defined as: low grade, with temperatures less than 200 °C; medium grade, with temperatures greater than 200 °C and less than about 600 °C; and high grade, with temperatures greater than 600 °C [14]. Most of the studies presented, and the current work, are focused on applications in low to medium grade heat resources.

Waste heat recovery is not a novel concept, as many studies have been conducted to demonstrate applicable technologies. During the early 1970s, the US Department of Energy funded research on applying an organic Rankine cycle to a Class 8 diesel engine truck [15–17]. Laboratory and highway tests showed a 12.5% improvement in the brake specific fuel consumption (the amount of fuel used over the power produced). This level of improvement was accomplished using an ORC that was not specifically designed for the application. In 1981, a diesel engine with Rankine cycle turbo-compounding was tested [18]. Fuel consumption improved by 14.8% when compared to the engine without the turbo-compound. More recent studies have focused on exploring different system configurations and cycle optimization. Hountalas et al. [19] experimented with the use of an exhaust gas recirculation cooler and a charge air cooler to increase the power output of the Rankine cycle. With an organic working fluid, the brake specific fuel consumption was improved by 11.3%. With water, the system showed a 9% improvement. Several authors have suggested using engine waste heat to produce cooling. Agnew and Talbi [20], [21] studied the combination of a

turbocharged diesel engine with an absorption refrigeration unit. Wang et al. [7], [22] recently demonstrated the use of a combined organic Rankine cycle and a vapor-compression cycle for diesel engine waste heat recovery. The results from these studies, and others, have shown that organic Rankine cycles are a promising technology for heat recovery.

Solar energy is a renewable resource that can be used to power the same systems that operate on waste heat. Some of the earliest studies involving the use of solar powered organic Rankine cycles were conducted in the late 1970s and early 80s [23], [24]. Recent advances in solar collector technologies have prompted many studies in the optimization of solar powered ORC systems [25–27]. Wang et al. [25] tested a 1.73 kW ORC system with an evacuated solar collector configuration and a flat plate solar collector configuration. They achieved overall system efficiencies of 4.2% and 3.2% for each case, respectively. Unfortunately, current solar power generation technologies tend to have high costs and complexity, which make it difficult to compete with cheaper alternatives. However, there are a few niche areas where solar energy is very marketable. One example of an application is the use of solar-thermal ORCs for reverse osmosis desalination [28–33]. Reverse osmosis desalination is a process of removing salts and minerals from water. This process is commonly used in areas of the world where the supply of fresh water is limited and sunlight is prevalent, such as the Middle East and parts of Southeast Asia. Studies have been conducted to improve performance and analyze the economic viability of these systems. Future studies will hopefully bring technologies that will allow solar-thermal power generation to be more reliable and affordable.

Another advanced and popular technology that is being investigated in conjunction with organic Rankine cycles is the thermoelectric generator (TEG). Thermoelectric generators

produce electricity in the presence of a temperature gradient, i.e. heat. TEGs are appealing because they are solid state, small, and lightweight. Several studies have modeled and successfully demonstrated waste heat recovery with the use of thermoelectric modules [34–39]. Miller et al. [38], [39] proposed and modeled a dual-cycle heat recovery system that couples the use of thermoelectric generators with an organic Rankine cycle. In Haidar and Ghojel's experiment [34], a temperature difference of 237 °C produced a maximum power of 12.2 W for one module. With 24 modules, Hsu et al. [36] produced a maximum power of 12.41 W at a 30 K temperature difference. Unfortunately, at this time, thermoelectric generators are expensive because they are made from special materials, and they do not output much power at low temperatures. If new materials or methods are discovered that will improve the power output and efficiency, TEGs could contribute significantly to waste heat recovery.

2.3 Heat Exchangers for Waste Heat Recovery

Heat exchangers are crucial to the success of any waste heat recovery system. The form of a heat exchanger is dominated by its application and working conditions. One type of heat exchanger that has been studied for heat recovery is the heat pipe heat exchanger. A heat pipe conducts heat transfer through liquid-vapor phase change. At the hot interface of the heat pipe, low pressure liquid is heated and vaporized in the evaporator section. The vapor travels to the cooler end of the pipe, then it condenses to back into a liquid, releasing thermal energy as it changes phase. The casing of the heat pipe is internally lined with a wicking material that allows the liquid to quickly return to the evaporator section via capillary force. The thermal cycle is illustrated in Figure 2.4.

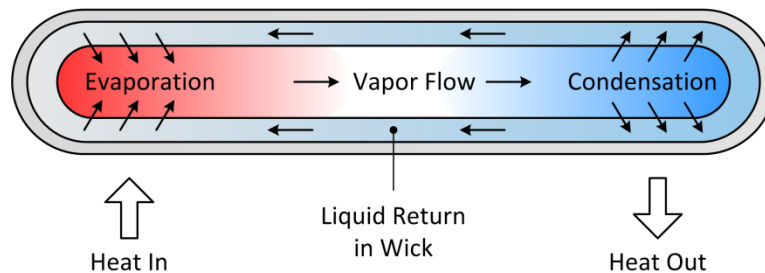


Figure 2.4. Heat pipe operation

A heat pipe heat exchanger (HPHE) consists of several finned heat pipes for heat transfer. HPHEs are capable of transporting a large amount of heat using a relatively small area. Also, power is not directly required for operation (however, power may be indirectly considered for fluids flowing over the pipes). Noie-Baghban and Majideian [40] designed and tested a HPHE for heat recovery in hospitals and laboratories, where fresh air is constantly circulated and thermally controlled. Yeng et al. [41] tested a HPHE for the purpose of recovering automotive exhaust gas to heat a large bus. Abd El-Baky and Mohamed [42] used a HPHE for precooling air into an air conditioning system. These three studies dealt with low-grade waste heat recovery, where the temperatures were less than 200 °C.

Another type of heat exchanger that has been used in heat recovery is the shell-and-tube heat exchanger. A diagram of a shell-and-tube heat exchanger is shown in Figure 2.5. Shell-and-tube heat exchangers consist of a bundle of tubes enclosed in a cylindrical shell. One fluid flows through the tubes, while another fluid flows around the tubes in the shell. There are many different configurations that have been developed and employed, depending on the application. Shell-and-tube heat exchangers are extremely versatile. They can be built to withstand pressures ranging from a vacuum to over 100 MPa, and temperatures from

cryogenic to over 1,000 °C [43]. These are the heavy duty heat exchangers in many industries, such as petroleum refinement, chemical processing, and power generation.

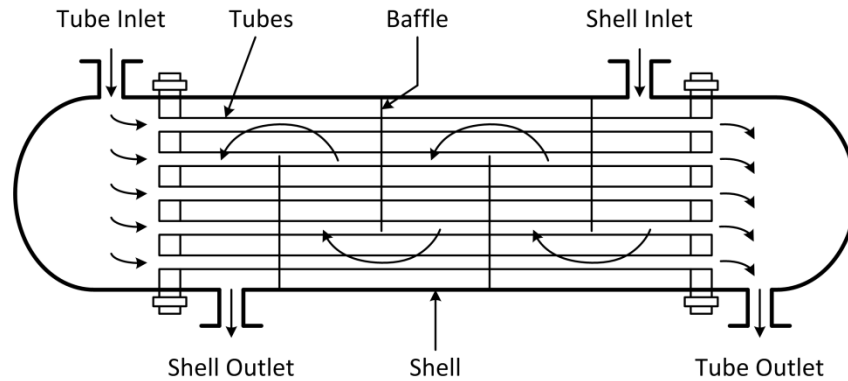


Figure 2.5. Shell-and-tube heat exchanger

A few studies have examined the potential of shell-and-tube heat exchangers for waste heat recovery applications. Pandiyarajan et al. [44] used a finned shell-and-tube heat exchanger in conjunction with a thermal storage system to recover heat from diesel engine exhaust. When the engine was at full load, their system was able to recover approximately 3.6 kW (15%) of the exhaust heat. Mavridou et al. [45] examined several shell-and-tube configurations for the sizing of an exhaust heat exchanger for truck applications. Using a diesel engine system with a Rankine bottoming cycle (the same used in [19]), they modeled the weight, volume, and pressure drop for designs with smooth, finned, and dimpled tubes. The heat output was held constant so sizing comparisons could be made. The authors found that the dimpled tube configuration only slightly reduced the weight, volume, and pressure drop of the heat exchanger when compared to the smooth tube design. For a finned tube arrangement, the weight was 51% less than the smooth tube arrangement, and the pressure drop was 45% less. However, the volume of the heat exchanger with the finned tubes was

slightly larger to accommodate the fins. The same study also compared the shell-and-tube heat exchanger configurations to several cross-flow plate-fin heat exchangers.

Plate-fin heat exchangers, particularly with a cross-flow configuration, are a good choice for mobile waste heat recovery applications because they can be constructed with low volumes and high surface areas. A plate-fin heat exchanger consists of corrugated fins or spacers sandwiched between plates to create flow passages. There are many available cross-sectional shapes, such as triangles, rectangles, and wavy designs. An example of a cross-flow heat exchanger with rectangular channels is shown in Figure 2.6.

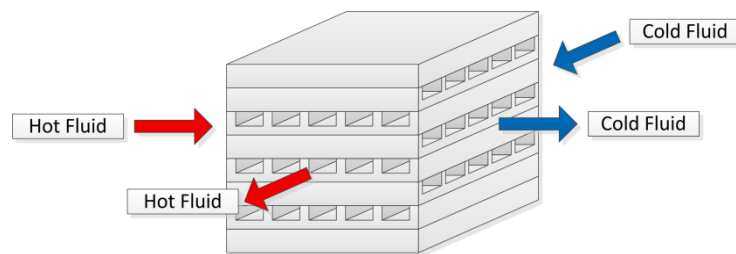


Figure 2.6. Single-pass, cross-flow plate-fin heat exchanger

The study by Mavridou et al. analyzed a single-pass, cross-flow plate-fin heat exchanger with triangular channels [45]. They also analyzed two configurations that combined the plate-fin arrangement with an open-cell metal foam with 10 pores per inch (ppi) and 40 ppi. Their computational results show that the 40 ppi metal foam and plate-fin design had the lightest weight and lowest volume for the same heat output when compared to all of the plate-fin and shell-and-tube configurations. However, the presence of the metal foam increased the pressure drop compared to the plate-fin heat exchanger with no foam.

Plate-fin heat exchangers can be even more compact and lightweight when constructed with microchannels. The following section introduces microchannel devices and their fabrication.

2.4 Microchannel Devices

2.4.1 Description of Microchannel Devices

The emergence of microchannel technology and microchannel devices comes from the development of integrated circuits. The demand for removing high heat fluxes using small devices was, and still is, the motivation for research into the possibilities of microchannel technology. Tuckerman and Pease were one of the first groups to demonstrate the potential of microchannel devices for very-large-scale integrated (VLSI) circuits [46]. They were able to show theoretically and experimentally that high power density circuit arrays could be adequately cooled with microchannel heat sinks. At the time, in 1981, it was believed that the limit for cooling, and thus the design of circuit arrays, was 20 W/cm^2 . The work of Tuckerman and Pease showed that, theoretically, $1,000 \text{ W/cm}^2$ was achievable, and they experimentally demonstrated sufficient cooling at 790 W/cm^2 . This was groundbreaking work, and it opened the door to the realm of microscale heat transfer.

Microchannels are generally classified as containing at least one feature dimension (e.g. diameter, height, or width) that is smaller than one millimeter. The general motivation behind microchannel devices can be illustrated by examining the definition of the Nusselt number. The Nusselt number is a nondimensional heat transfer parameter that relates the convective and conductive heat transfer mechanisms in fluid flow. It is defined as

$$Nu = \frac{hD_h}{k} \quad (1.1)$$

where h is the convective heat transfer coefficient for the fluid, D_h is the hydraulic diameter of the flow area, and k is the fluid thermal conductivity. The hydraulic diameter is defined as

$$D_h = \frac{4A}{P} \quad (1.2)$$

where A is the channel cross-sectional area and P is the channel perimeter. For fully developed laminar flow in a pipe with either a constant surface heat flux or constant surface temperature, the Nusselt number is a constant [47]. Therefore, as the hydraulic diameter decreases, the convective heat transfer coefficient increases. A high convective heat transfer coefficient directly increases the amount of heat removal, according to Newton's law of cooling

$$Q = hA\Delta T \quad (1.3)$$

For a constant area A and constant temperature gradient ΔT , an increase in h increases the heat transfer Q . This is why microchannels are appealing in heat transfer applications.

Another desirable aspect of microchannel devices is the high surface area to volume ratio. This translates to considerably smaller package sizes without the loss of heat transfer capability. Additionally, the volume of working fluid necessary for operation is less than conventionally sized heat exchangers.

A nontrivial consideration of microchannel devices is the fluid pressure drop. For a single round channel, smaller diameters result in higher pressure drops if the length is held constant. This can be explained mathematically by looking at the pressure drop equation for fully developed flow in a pipe.

$$\Delta P = f \frac{\rho v^2}{2} \frac{L}{D} \quad (1.4)$$

where the pressure drop ΔP is a function of the Moody (or Darcy) friction factor f , the fluid density ρ , fluid velocity v , channel diameter D , and channel length L . The pressure drop along a channel is inversely proportional to the diameter, therefore a decrease in diameter results in an increase in the pressure drop when all else is constant. A high pressure drop directly increases the pumping power required for operation.

When considering an array of multiple channels, it is possible to design a microchannel device that, theoretically, suffers no penalty in pressure drop in comparison to a conventionally sized one. For a constant total cross-sectional flow area, a reduction of the channel diameter means more channels can occupy the device. If the microchannel device is shortened, the pressure drop could be the same as a conventional device. Pressure drop and back pressure regulation are important for diesel engine waste heat recovery with microchannel devices. Excessive back pressure will negatively affect engine performance, which is counterproductive.

2.4.2 Fabrication of Microchannel Devices

For metals, particularly stainless steel and titanium, one of the most common methods of fabricating microchannel devices is photochemical etching followed by diffusion bonding. Photochemical etching is a process in which material is chemically removed from a substrate, or shim, to form a desired pattern. Typical shim thicknesses range from about 0.1 mm to 2 mm. Understanding the etching process is important to this study because it is directly tied to

the measurement results. The process of photochemical etching is illustrated in Figure 2.7 and explained here.

Photochemical etching starts with the lamination of a photoresist onto the top and bottom of a shim. The photoresist contains all of the features that will be etched on the shim. There are two types of photoresists: positive and negative. A positive photoresist will soften when exposed to ultraviolet (UV) light, whereas a negative photoresist will harden when exposed. To print the design pattern onto the photoresist, a photomask is placed on the photoresist in areas where the photoresist is to be kept (positive) or removed (negative). After development with UV light, the photomask and any soft areas of the photoresist are washed away in a solution. What remains is the plasticized photoresist and areas of exposed metal substrate. Next, the chemical etchant is applied. Areas of exposed substrate are eaten away, while areas covered by the photoresist are maintained. Once etching is complete, the remaining photoresist is removed, leaving behind the metal substrate with the etched design pattern. More information about photochemical etching can be found in [48], [49].

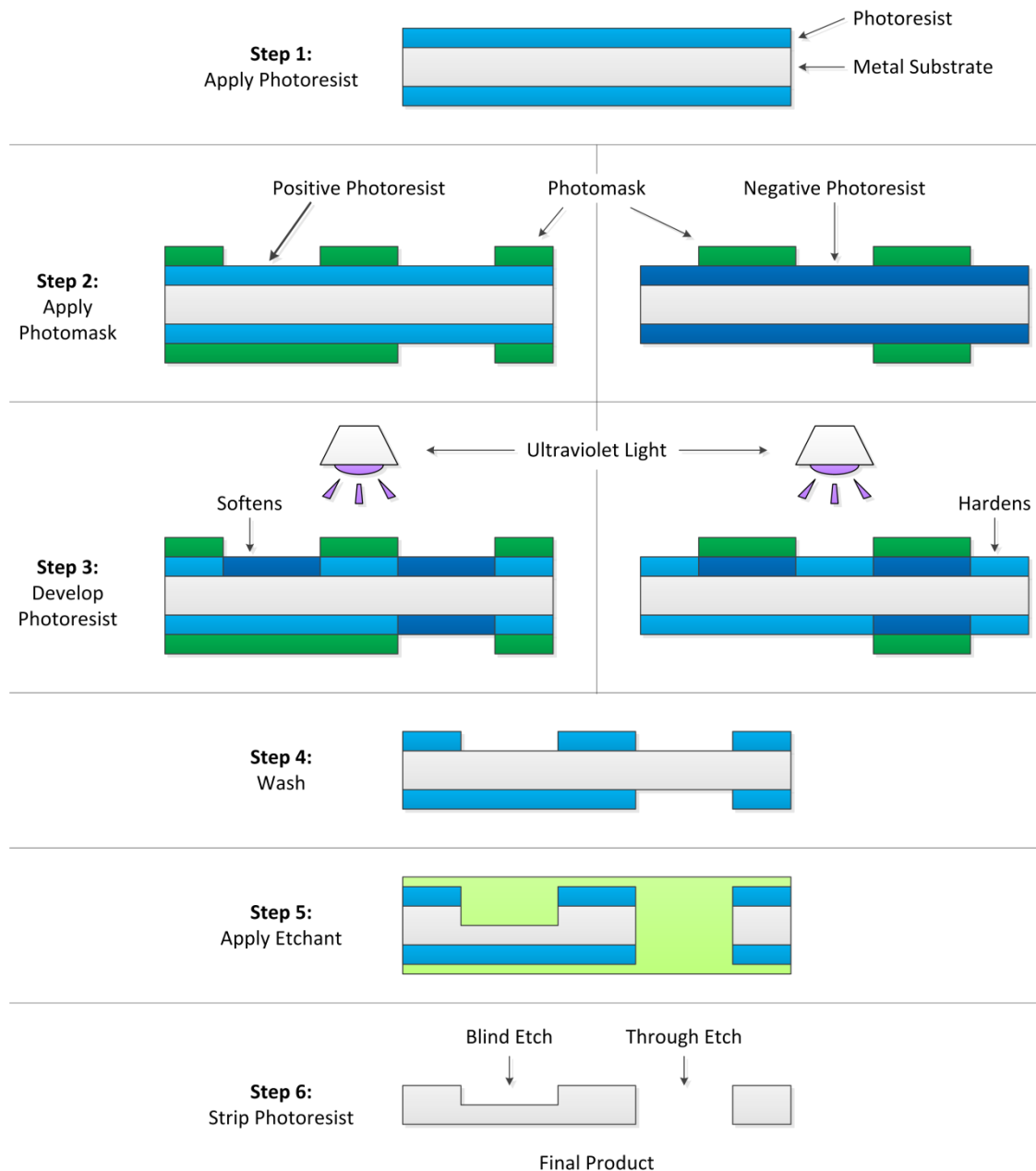


Figure 2.7. Photochemical etching process (with positive or negative photoresist)

Photochemical etching can produce profile artifacts that may be undesirable in the finished part. For blind etches, corners will be rounded instead of square. The corner radius is

mostly dependent on the etch depth and etch rate. For through etches, it is possible to have bicuspid edges form where the etchant breaks through the part, as in Figure 2.8.

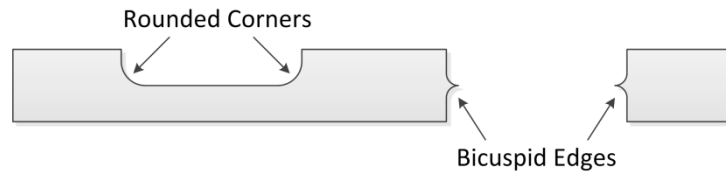


Figure 2.8. Photochemical etching profile artifacts

Diffusion bonding and diffusion brazing are two ways of assembling a stack of etched shims to form a microfluidic device. In diffusion bonding, the stack is put into a vacuum or inert gas chamber where it is pressed at high temperature and pressure. This process can be at temperatures up to two-thirds of the material melting point, and usually takes several hours [50]. Diffusion bonding allows atoms from the mating parts to migrate across the contact surface to form a continuous bond. In diffusion brazing, an interstitial, or filler, layer is added between the faying components to help facilitate the bonding process. Diffusion brazing typically requires lower temperatures and pressures than diffusion bonding to achieve a complete bond because of the interstitial layer.

2.5 Microchannel Profiling and Imaging

There are generally two classifications of methods that can be employed to measure microchannels. The first classification is destructive testing. In many designs, the microchannels cannot be accessed unless the device is cut or parted. Generally, this results in destroying the device such that it is unusable afterward. Taking measurements of a device that has been cut can be challenging because it is possible that the cutting operation may deform

the channels. The second classification is nondestructive testing. If the channels are optically accessible, then measurements can be taken without damaging the device.

Two of the most common instruments that are used to measure microchannels are the scanning electron microscope (SEM) and the profilometer [51]. An SEM analyzes how a beam of electrons interacts with a sample surface. When using an SEM, the sample surface should be electrically conductive. For most metals, this is not a significant issue. For other materials, a thin coating of a conductive material is usually applied before measurements are taken. An optical or contact profilometer may also be used to measure the dimensions of a microchannel. Optical profilometers detect the interference patterns of light that reflects from a sample surface. This is known as interferometry. Contact profilometers physically touch the sample surface with a stylus that moves along a path to produce a profile. Three-dimensional optical profilometry is a widely used method for measuring surface roughness.

There are a few studies that mention measuring or profiling microchannels. Choi et al. [52] used both an SEM and optical profiler to measure the dimensions of a rectangular microchannel for a study in surface wettability. Yeong et al. [53] used a profilometer and confocal microscope to profile rectangular microchannels in a falling film reactor.

3 Heat Exchanger Design

3.1 Waste Heat Recovery System

The Heat Recovery Unit is one component of a waste heat recovery system. The system employs a combined organic Rankine cycle and a vapor-compression cycle build by Wang et al. [7]. It is designed to produce cooling from the exhaust of a diesel engine. A schematic of the system is shown in Figure 3.1.

The waste heat recovery process starts at the diesel generator. Exhaust gases from the diesel generator pass through the Heat Recovery Unit and provide energy to the heat transfer oil. The oil is pumped through the boiler of the organic Rankine cycle, energizing the working fluid before it enters the expander. In this system, the expander is directly coupled to the compressor of a vapor-compression cycle to produce refrigerated air. Notable technologies utilized in the combined cycle include a small-scale scroll expander and scroll compressor, and an integrated microchannel boiler-recuperator.

The performance of the combined ORC and vapor-compression cycle has been modeled and tested by Wang et al. [7]. In their tests, an oil circulator was used in place of the diesel generator to simulate waste heat at 200 °C. At its design point, the system was expected to produce 5.3 kW of cooling. Under laboratory test conditions, the combined cycle produced 4.4 kW of cooling with an overall system coefficient of performance (COP) of 0.48. The working fluids were HFC-245fa for the organic Rankine cycle and HFC-134a for the vapor-compression cycle. The heat transfer oil used in the oil circulator, and the one used with the Heat Recovery Unit in this study, was Paratherm NF.

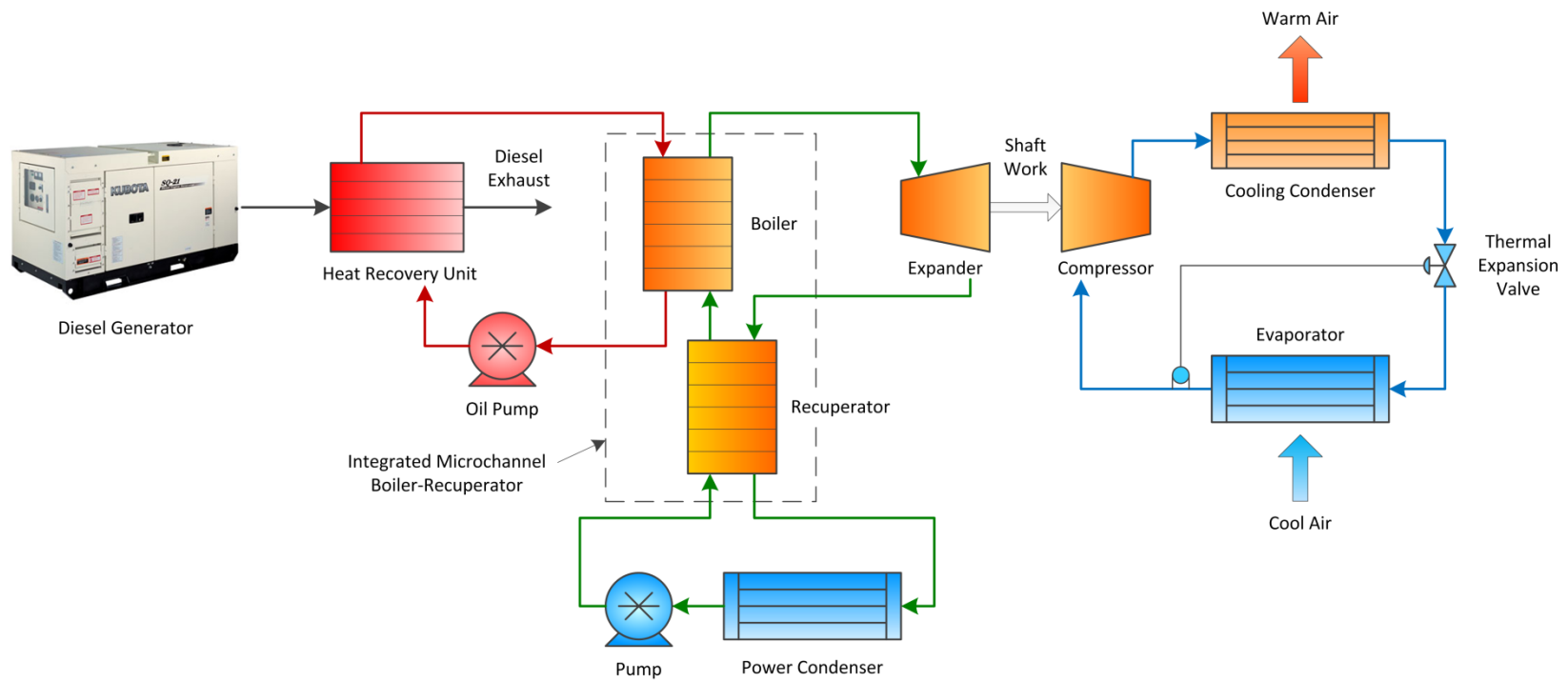


Figure 3.1. Waste heat recovery system

3.2 Heat Recovery Unit Design

The Heat Recovery Unit is a two-pass, cross-counter flow microchannel heat exchanger. A schematic of this type of heat exchanger is shown in Figure 3.2 and a SolidWorks model of the Heat Recovery Unit is shown in Figure 3.3.

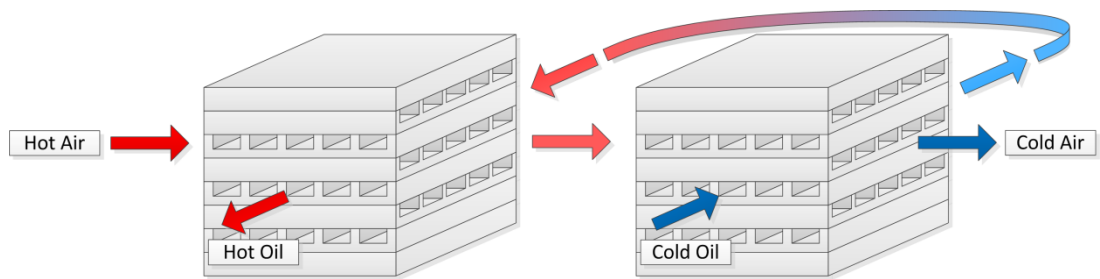


Figure 3.2. Two-pass, cross-counter flow plate-fin heat exchanger

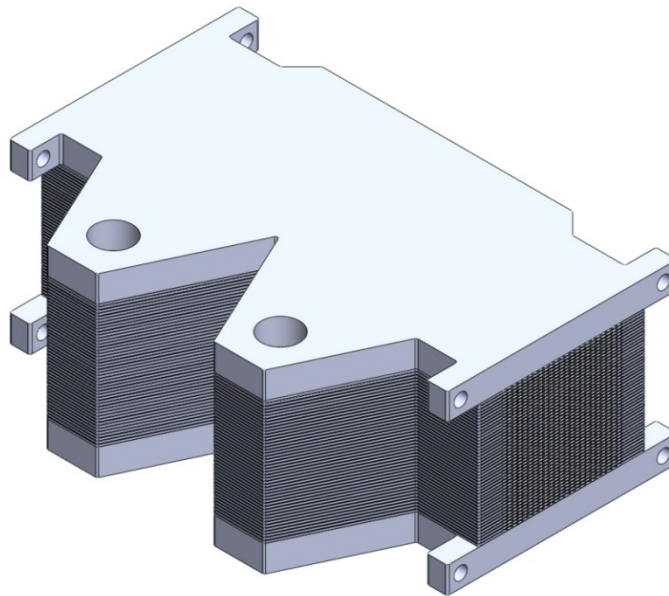


Figure 3.3. SolidWorks model of the Heat Recovery Unit

The HRU is designed to fit within the chassis of the Kubota SQ-14 diesel generator. The major dimensions of the HRU are 210 mm (8.3 inches) in length, 145 mm (5.7 inches) in

width, and 85 mm (3.4 inches) in height. The Heat Recovery Unit is a diffusion brazed device that consists of a stack of alternating stainless steel shims. The shim stack is sandwiched between two plates that contain mounting and manifold features, as well as 1/2 inch NPT tapped holes for the oil ports.

There are two different shim geometries in the Heat Recovery Unit. The first is the exhaust shim (also referred to as the air shim) and the second is the oil shim. Both shim types are designed with rectangular microchannels and are produced by photochemical etching. The exhaust shim, shown in Figure 3.4, features channels that run the entire length of the shim.

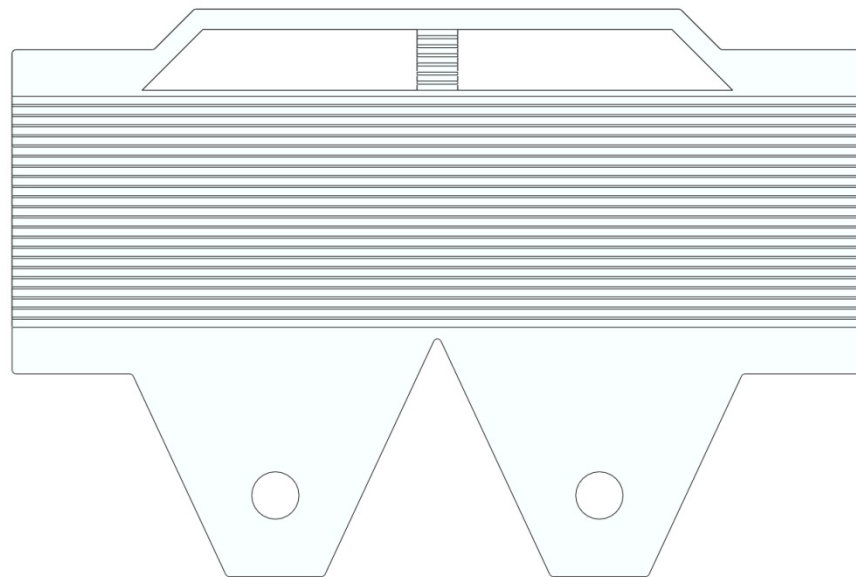


Figure 3.4. Exhaust shim

The exhaust channels are straight, with relatively wide cross-sectional area to reduce the back pressure on the engine and the potential of soot buildup from the diesel exhaust. Exhaust gases enter and leave through manifolds attached at the two ends of the heat exchanger. The oil shim, depicted in Figure 3.5, contains two sections of channels that make up the two-pass flow

arrangement. Flow distribution veins help guide the oil from the inlet port to the channels of the first pass. After the oil has passed through the first set of channels, it enters a plenum where it can mix with the oil from other layers before going through the second pass. The mixing plenum exists to improve the temperature uniformity going into the second pass.

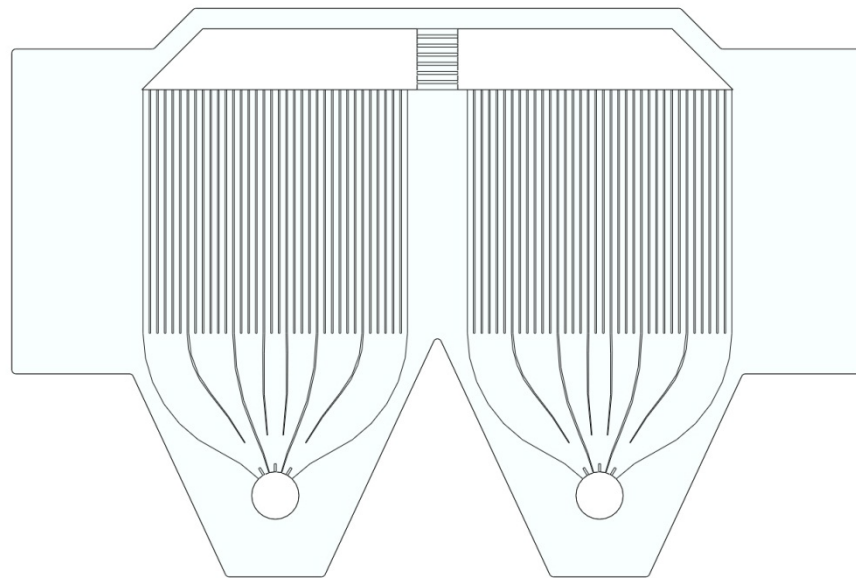


Figure 3.5. Oil shim

Table 3.1 lists the channel dimensions of both shims, as well as the number of shims present in the complete device. The channel dimensions and number of channels on each shim were selected from a model that optimized the heat transfer. The core of the Heat Recovery Unit has a surface area of 0.773 m^2 and a volume of $2.21 \times 10^{-4} \text{ m}^3$ for the air stream, and 0.624 m^2 and $4.25 \times 10^{-5} \text{ m}^3$ for the oil stream. The model used to determine the dimensions is discussed in the next section.

Table 3.1. Channel dimensions and information of the shims

	Exhaust Shim	Oil Shim
Number of Channels	23	35 (per pass)
Channel Length (mm)	210	60
Channel Width (mm)	2.00	1.50
Channel Depth (mm)	0.80	0.15
Shim Thickness (mm)	0.99	0.30
Number of Shims in the Bonded Device	46	45

3.3 Thermal Modeling

The Heat Recovery Unit was designed to operate as an intermediary between the diesel generator and the Rankine cycle. As such, there were physical and thermal considerations that influenced the design. Physically, the size of the HRU was limited by the space available within the chassis of the diesel generator. Thermally, the HRU had to work within the operating conditions of the diesel generator and organic Rankine cycle. Other factors, such as soot deposition and pressure drop, were also considered in the design.

Based on the expected 5.3 kW cooling capacity of the vapor-compression cycle and an overall system COP of 0.5, the Heat Recovery Unit was designed to recover at least 10.6 kW of heat from the diesel engine. A thermal model was developed in MATLAB[®] [54] to optimize the heat transfer and geometry of the HRU to achieve this goal.

The Heat Recovery Unit was modeled as two independent single-pass cross-flow heat exchanger sections. The two passes were linked to each other by the intermediate air and oil temperatures. An illustration of the model is presented in Figure 3.6.

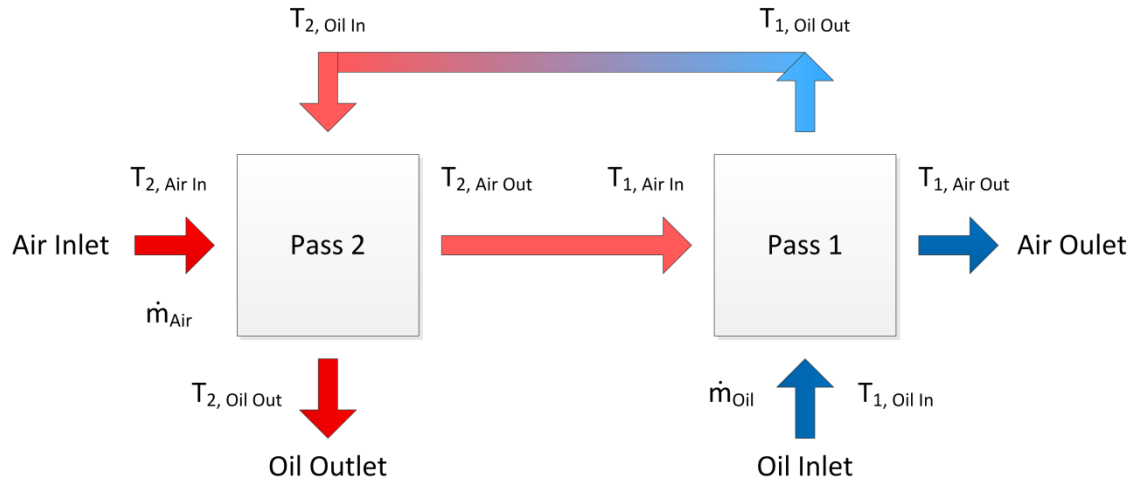


Figure 3.6. Thermal model diagram

For given mass flow rates and inlet temperatures of air and oil, the model iteratively solved the heat transfer of the two passes until the intermediate temperatures were consistent (i.e.

$$T_{1, \text{Air In}} = T_{2, \text{Air Out}} \text{ and } T_{1, \text{Oil Out}} = T_{2, \text{Oil In}}).$$

A list of the major assumptions in the model is provided below.

Model Assumptions

- Uniform flow (velocity and temperature)
- Constant, uniform, and rectangular cross-sectional channel areas
- Negligible heat loss
- Negligible axial conduction
- Laminar, fully developed flow
- No entrance effects (thermal or hydraulic)
- Temperature dependent properties (evaluated at mean temperatures)
- Uniform channel surface temperature

Calculations of the heat transfer and pressure drop were performed using the channel regions in the core of the heat exchanger. For the exhaust shim, the channel area for heat transfer was based on the length of channel that was directly above the oil cross-flow sections. The pressure drop for the exhaust shim was calculated using the full shim length. For the oil shim, the channel length for both heat transfer and pressure drop was based on only the

straight channel region. This does not include the effects of the header ports, flow distribution veins, or the mixing plenum. Figure 3.7 shows the model regions of both shims.

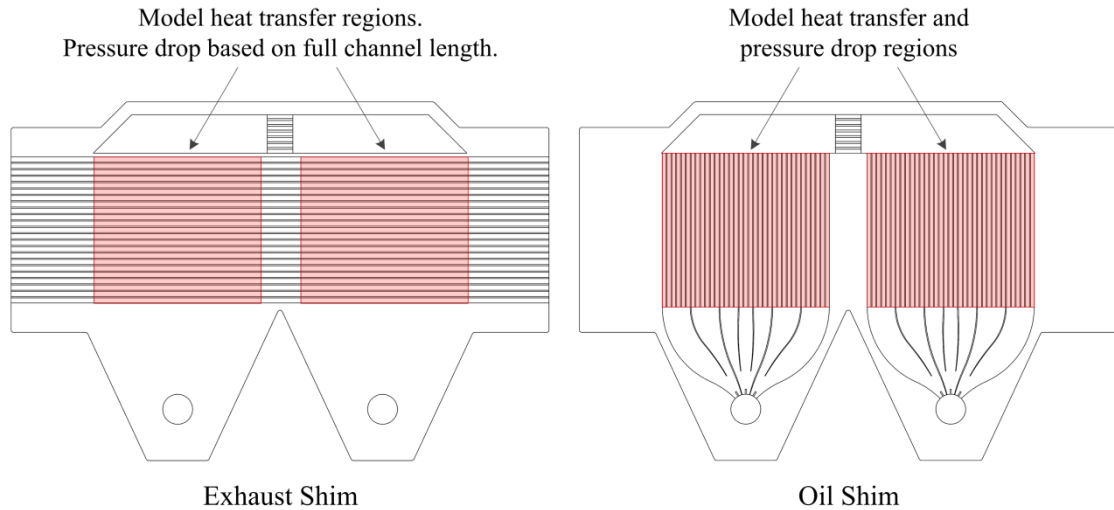


Figure 3.7. Model heat transfer and pressure drop regions for both shim geometries

Using this model, the heat transfer performance was calculated for various channel geometries. The final channel geometries were presented in Table 3.1 of the previous section. Table 3.2, below, shows the thermal model results for the design conditions and geometry.

Table 3.2. Heat Recovery Unit design conditions and model results

Input Parameters		Model Results	
Air Inlet Temperature (°C)	488	Air Outlet Temperature (°C)	118
Oil Inlet Temperature (°C)	100	Oil Outlet Temperature (°C)	200
Air Flow (g/s)	30	Duty (kW)	11.6
Oil Flow (g/s)	46	Effectiveness	0.95
		Air Pressure Drop (kPa)	2.0
		Oil Pressure Drop (kPa)	22.0

The Heat Recovery Unit should recover approximately 11.6 kW of thermal energy; enough to run the combined ORC and vapor-compression cycle. For more information and detailed discussions on the design and thermal modeling of the Heat Recovery Unit, see [55].

3.4 Fabrication Results

Photochemical etching of the Heat Recovery Unit was done by VACCO Industries and the diffusion brazing was by Vacuum Processing Engineering, Inc. Figure 3.8 and Figure 3.9 show pictures of the fabricated device. Photos of extra shims provided by VACCO are shown in Figure 3.10 and Figure 3.11. Unfortunately, the shims were slightly damaged during shipment. Upon examination of Figure 3.9, there are three significant points to note about the actual device. First, the exhaust channels were not completely rectangular as in the design. Instead, the channels had a curved bottom surface as a result of the etching process. Second, the channels appear to be inconsistent in size, even to the naked eye. Third, it is visually obvious that the channels were misaligned. These notes will be discussed in detail later in this study.

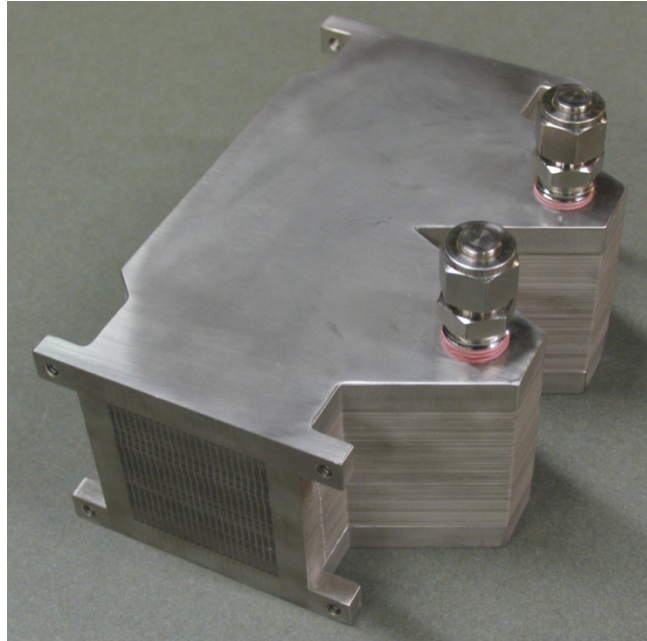


Figure 3.8. A photograph of the Heat Recovery Unit

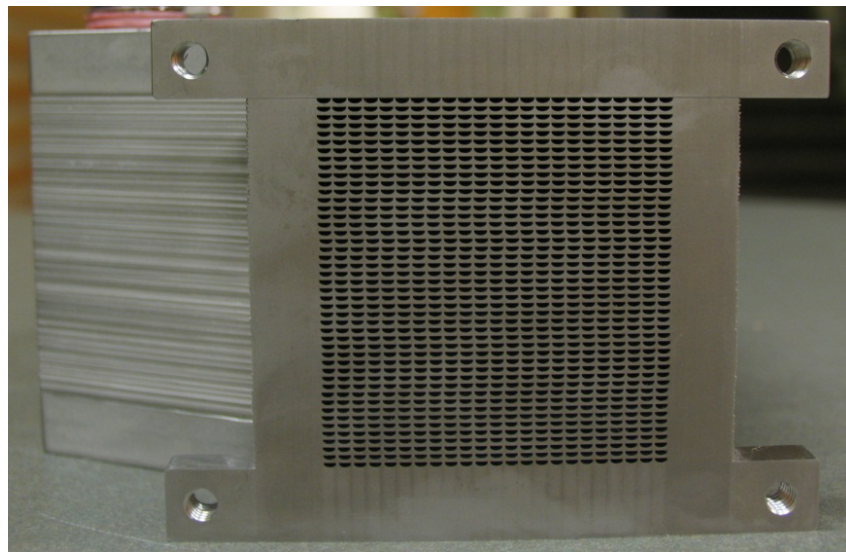


Figure 3.9. A photograph of the exhaust channels

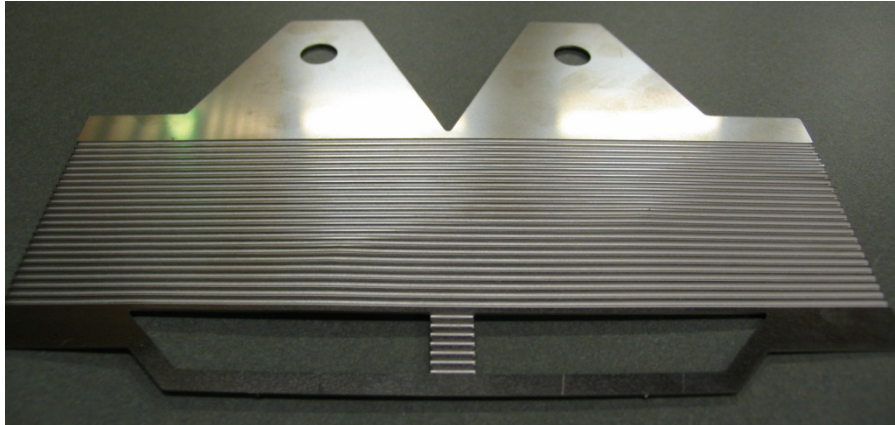


Figure 3.10. Exhaust shim photograph

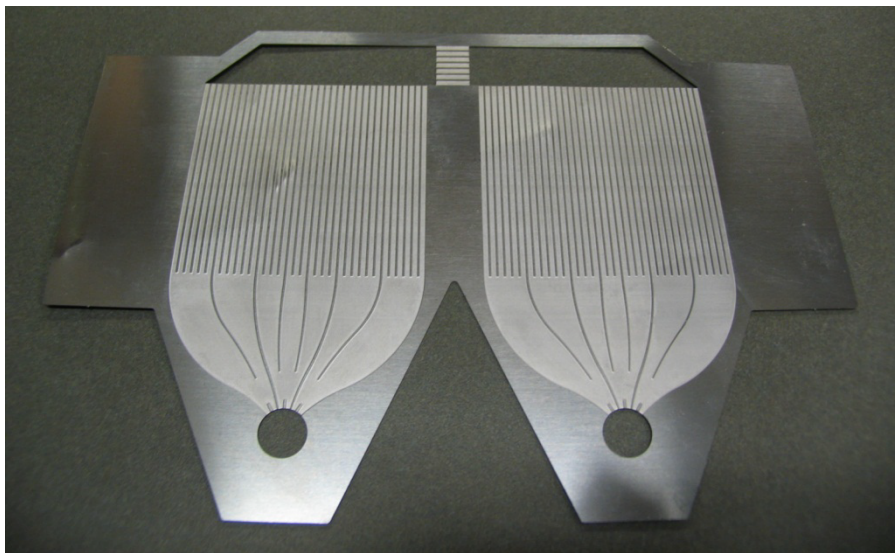


Figure 3.11. Oil shim photograph

4 Scope and Objectives

The scope of this study is to assess the manufacturing accuracy and thermal performance of the Heat Recovery Unit. The primary objectives are to measure the channel dimensions of both the exhaust and oil shims; determine the thermal performance using heated air to simulate diesel exhaust; and measure the velocity and temperature profiles of the outlet exhaust channels. Examining the physical dimensions of the exhaust and oil channels will be helpful in identifying the manufacturing accuracy and may be useful in interpreting the thermal performance results. The thermal performance of the Heat Recovery Unit will be measured by the classic thermodynamic quantities such as duty, effectiveness, and pressure drop. These parameters, along with the velocity and temperature distributions, will be used to evaluate the validity of the theoretical model and the design. Conclusions from this study will help improve the design of future generations of the Heat Recovery Unit.

5 Heat Exchanger Testing

In this chapter, the methods of testing, calibration, and the facilities used to accomplish these tasks are described. Testing was divided into two categories that coincide with the primary objectives of this study: manufacturing tests and thermal tests. The manufacturing tests consisted of the Microscope Image Analysis and the Shim Profile Analysis. The purpose of these two analyses was to verify that the manufacturing process produced the intended microchannel geometries. If the fabricated results were different from the design, then the actual geometry would be related to the heat transfer performance. The thermal testing category contains the thermal performance tests and the traverse measurements. For this study, the thermal tests were conducted with heated air to simulate diesel engine exhaust. The traverse measurements were designed to measure the velocity and temperature profiles of the exhaust channels. Sections 5.1 and 5.2 concentrate on the manufacturing testing, while Section 5.3 focuses on the thermal testing.

5.1 Microscope Image Analysis

5.1.1 Microscope System

For the Microscope Image Analysis, the goal was to measure the area of the exhaust channels on the Heat Recovery Unit. For this task, a Leica Wild M3Z stereoscopic microscope was used because it was mounted on a boom stand that could accommodate the size of the HRU. A Sony 3CCD DXC-960MD camera was attached to the microscope and transmitted images to a computer. A Boeckeler VIA-100 video measurement system was installed between the microscope and computer. The VIA-100 provided adjustable overlay marks and

measurement scales. The Boeckeler video measurement system was used for calibrating the microscope and collecting measurement images. Further analysis was performed on the images using MATLAB. The analysis is detailed in Section 6.1.1. Images from the microscope were saved using FlashBus Spectrim FBG 1.2 software. A photograph of the microscope testing facility is shown in Figure 5.1.

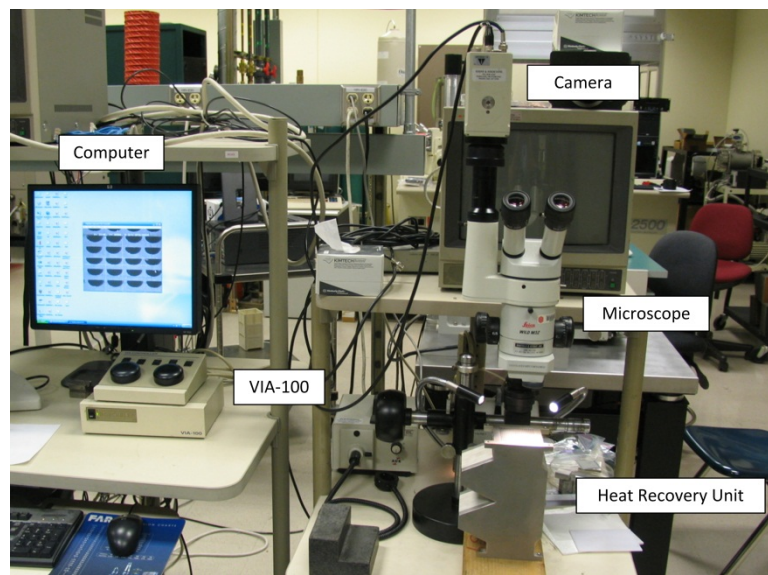


Figure 5.1. A photograph of the microscope testing facility

5.1.2 Microscope Calibration

Before images of the exhaust channels were captured, the measurement system was calibrated for length measurements at the available magnification levels. A NIST traceable stage micrometer (model KR-812) was used in conjunction with the VIA-100 system to set the length scale for each magnification. Figure 5.2 is an example image taken at 10x of the calibration standard and the measurement overlay of the VIA-100. This picture was used to calibrate the x-direction. Each dashed line on the overlay was two pixels wide in its smallest

dimension. The length to pixel conversion was obtained by counting the number of pixels in a straight line between the centers of two dashed lines.

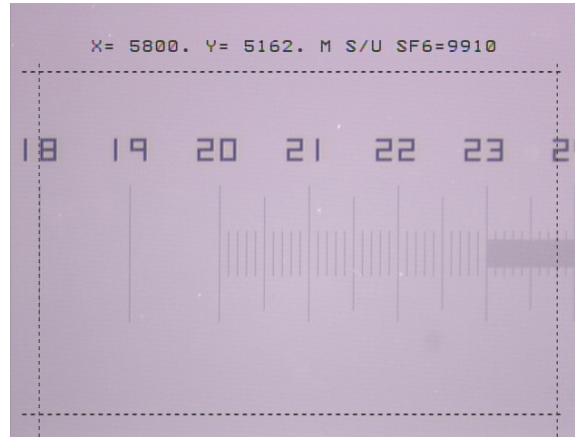


Figure 5.2. Example calibration image for the x-direction at 10x

A consideration that was recognized during the calibration and measurement process was the principle by which the images were captured. Stereomicroscopes use two objectives that view the sample at different angles to produce a three-dimensional image. Since the CCD camera effectively looks through one eyepiece of the microscope, the image projected onto the screen is at an angle to the sample. This could potentially create an error in the length measurements if the scaling did not account for the angle. By calibrating the length scales using the overlays on the camera image, the angular effects are much reduced, or essentially eliminated, from the measurement process.

5.1.3 Measurement Process

A preliminary image of the exhaust channels was saved in order to verify the calibration and general channel size. From this image, in Figure 5.3, the channel measurement appears to

be reasonably close to its design. The reference image also shows debris in the channels. Care was taken to remove most of the debris before the usable images were saved.

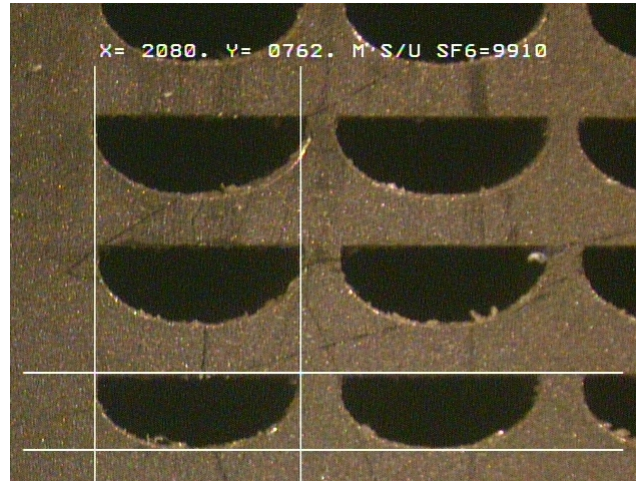


Figure 5.3. Microscope reference image at 10x

At a magnification of 10x, all 1,058 exhaust channels were fully captured for one side of the heat exchanger (six channels per image for a total of 192 images). The images were saved as jpeg formatted files without the VIA-100 overlay. At this magnification, each pixel is 10 μm x 10 μm .

5.2 Shim Profile Analysis

5.2.1 ZeScope Optical Profilometer

The purpose of the Shim Profile Analysis was to measure the channel profiles of the exhaust and oil shims. For these tests, a ZeScope Optical Profiler was used to analyze the shims. The ZeScope came with two pieces of software. The first program was aptly named ZeScope, and it was the interface to the profiler. With this program, users can measure profiles, visualize results, and perform various surface metrology analyses. The second

software was the Scanning Probe Image Processor, SPIP™. This program can perform complex surface metrology analyses on the profiles acquired by the ZeScope. Both programs were used in the profilometer analysis. Their use is detailed in Section 6.1.2. The RMS repeatability of the ZeScope surface measurements was 0.01 nanometers.

5.2.2 Measurement Process

Profiles were acquired at 5x magnification. On the exhaust shim, all twenty-three channels were profiled near the edge of the shim. On the oil shim, all thirty-five channels of the first pass were profiled. Locations of the profile regions are illustrated in Figure 5.4. These regions were chosen because they had the least amount of damage or warping from shipping. To reduce shim bow, weights were placed on the shims during profiling.

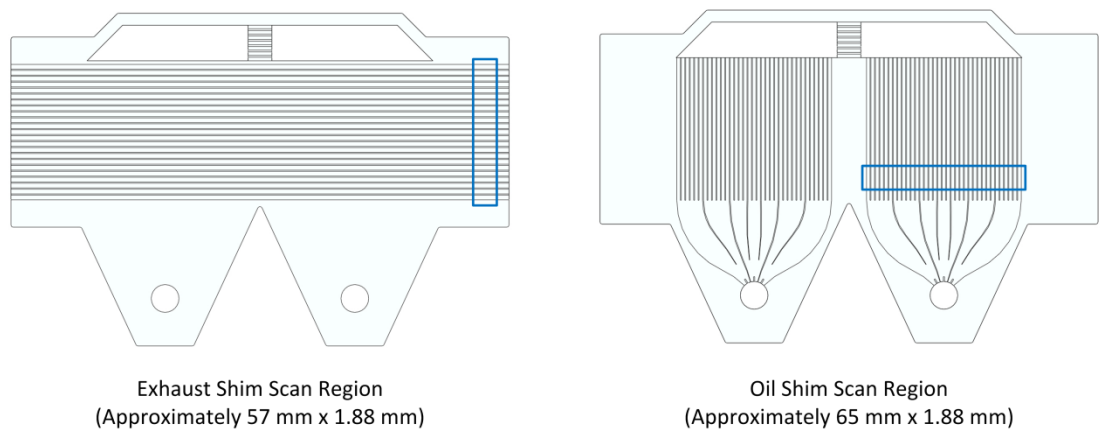


Figure 5.4. Profilometer scan regions

Scanning regions were partitioned into sections for each shim. This was done so the profiles would be saved in batches, rather than all at once, to avoid losing the data in case of an error. For the exhaust shim, there were six scans with four full channels in each scan, except for the last which had three. For the oil shim, there were seven scans with five full

channels in each scan. A scan of the oil shim is shown in Figure 5.5. Some post processing was done after scanning the shims. Operations such as filling in the void pixels and reducing noise via averaging were applied to all scans. After obtaining the 3-D profiles of the shims using the ZeScope software, SPIP was used to collapse the data into 2-D profiles for further processing in MATLAB.

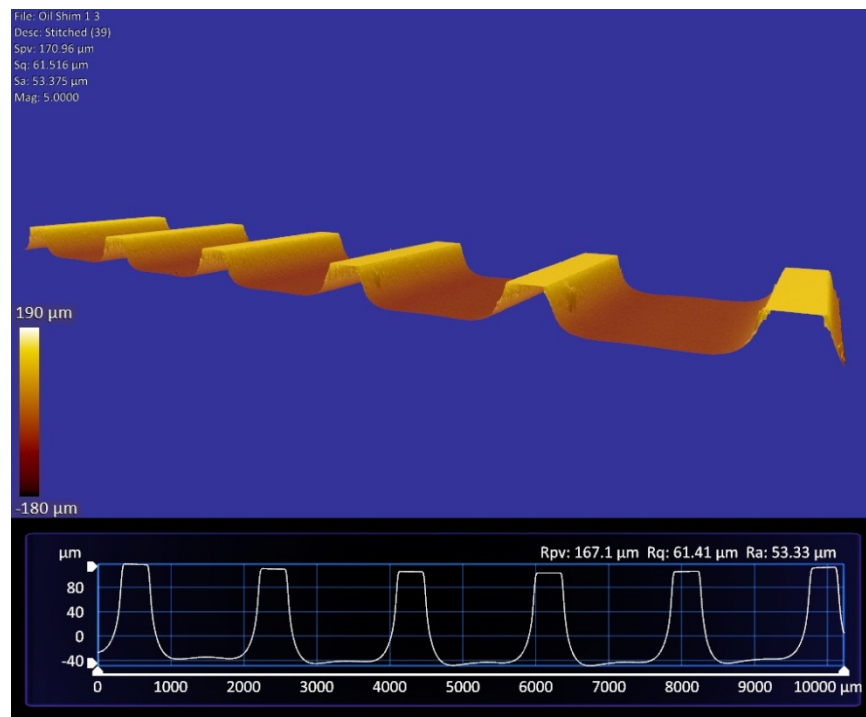


Figure 5.5. ZeScope screenshot of oil shim channels and their profiles

5.3 Thermal Testing Facility

5.3.1 Test Stand and Components

The thermal testing facility had the capacity to test the thermal performance and measure velocity and temperature profiles of the Heat Recovery unit using hot air in place of diesel exhaust. A process and instrumentation diagram for the thermal testing facility is presented in Figure 5.6.

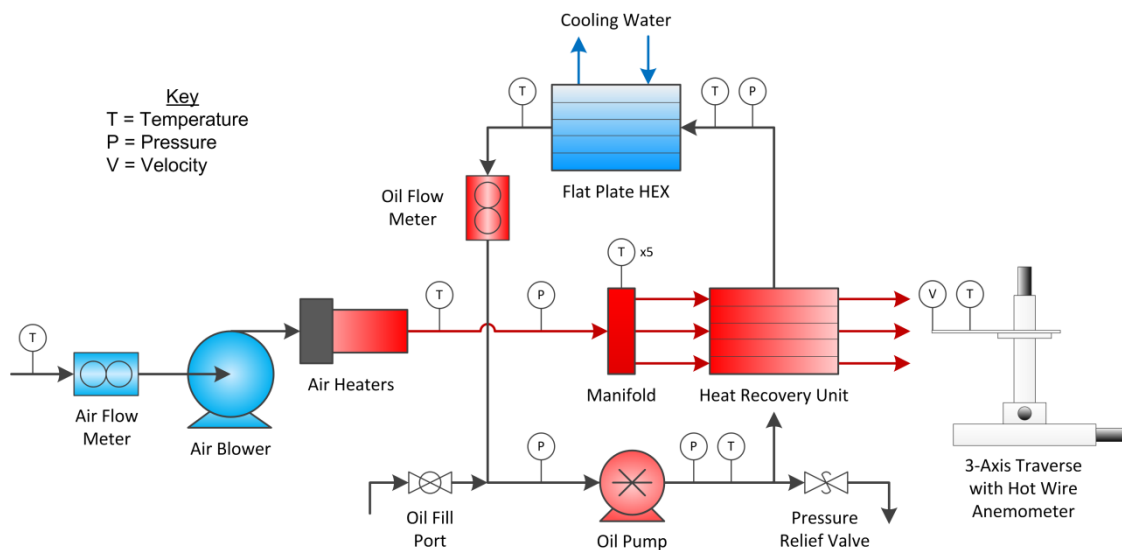


Figure 5.6. Heat Recovery Unit thermal test bench schematic

A regenerative blower intakes ambient air and passes it through two electrical heaters. The hot air then enters a manifold where it is allowed to distribute before entering the exhaust channels of the Heat Recovery Unit. After the air exchanges heat with the oil in the HRU, it is exhausted into the ambient environment. A gear pump circulates oil through the HRU and into a flat plate heat exchanger. Cooling water runs through the flat plate heat exchanger to remove heat from the oil.

A custom built manifold was made for ducting the hot air into the Heat Recovery Unit. The manifold featured five holes for thermocouples to measure the air temperature distribution before it enters the heat exchanger. A pressure port was also included on the side. Figure 5.7 shows a picture of the manifold before it was attached to the HRU. A graphite gasket was used to seal the mating faces of the HRU and manifold. The outlet of the HRU was exposed to the ambient air.

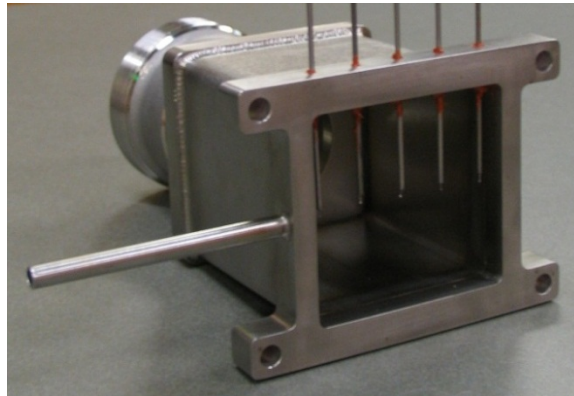


Figure 5.7. Manifold photograph

The instrumentation of the test bench included a hot wire anemometer, K-type thermocouples, solid state pressure transducers, and turbine flow meters. A 3-axis linear traverse system was used to collect temperature and velocity measurements on the exhaust outlet of the Heat Recovery Unit. A single probe hot film anemometer (often referred to as a hot wire in this document) was mounted to the LabVIEW controlled traverse system. The film had a sensing diameter of 25.4 μm and a length of 0.25 mm (recall that the exhaust channels were designed to 2 mm by 0.8 mm). A 1/16-in (~ 1.6 mm) diameter thermocouple was attached to the probe support. LabVIEW was used for the data acquisition and traverse control. A picture of the thermal testing facility is shown in Figure 5.8. Equipment and

instrumentation specifications can be found in Table 5.1 and Table 5.2. Measurement uncertainties are presented in Sections 6.2.3 and 6.2.4.

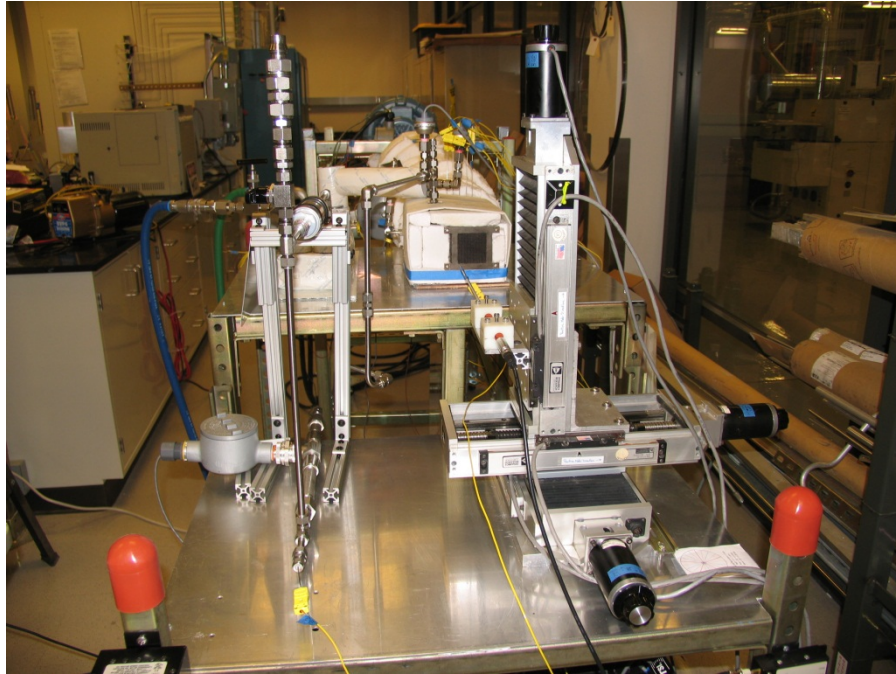


Figure 5.8. Thermal testing facility

Table 5.1. Equipment information and specifications

Description	Manufacturer	Model	Range	Qty
Air Blower	AMETEK	DR505CD72M	160 scfm max	1
Air Heater	Sylvania	SuperHeat MAX F074728	10 kW	2
Oil Pump	LiquiFlo	43S6PEEN20C	1.45 gpm (5.48 lpm)	1
Flat Plate Heat Exchanger	FlatPlate	FP5X12-20	20 Plates 4.9 in x 12.2 in x 2.2 in	1
Linear Stage	Parker	406006LN	6 in (150 mm)	3
Motor Driver	CompuMotor	SX6	-	2

Table 5.2. Instrumentation information and specifications

Description	Manufacturer	Model Number	Range (Selected)	Accuracy	Qty
Hot Wire Anemometer	TSI	Model 1750 (System) 1260A-10A (Probe)	-	-	1
Turbine Flow Meter	OMEGA	FTB-938	8-130 acfm	1% Reading 0.25% Repeatability	1
Flow Meter Signal Conditioner	OMEGA	FLSC-61	75-375 Hz (350-1950 Hz) 1875-10000 Hz	0.3% FS Linearity	1
Turbine Flow Meter	OMEGA	FTB-901T	0.5-2.5 gpm	0.5% Reading 0.05% Repeatability	1
Flow Meter Signal Conditioner	OMEGA	FLSC-62A	(100-1000 Hz) 1 KHz-10 KHz	0.3% FS Linearity	1
Pressure Transducer	Cole-Parmer	68075-18	0-100 psig	0.25% FS	2
Pressure Transducer	Cole-Parmer	68075-16	0-50 psig	0.25% FS	2
Pressure Transducer	OMEGA	PX209-030A5V	0-30 psia	0.25% FS	1
Thermocouple	OMEGA	KMQSS-062U-6 KMQSS-062E-6	-200 to 1250 °C	Greater of 2.2 °C or 0.75%	11
Data Acquisition Chasis	National Instruments	NI cDAQ-9178	-	-	1
Thermocouple Input Module	National Instruments	NI 9213	16 Ch, 1,200 S/s	-	1
Analog Voltage Input Module	National Instruments	NI 9201	8 Ch, ±10 V 12-Bit, 500 kS/s	-	1
Analog Current Input Module	National Instruments	NI 9203	8 Ch, ±20 mA 16-Bit, 200 kS/s	-	1

5.3.2 Thermal Test Matrix

The testing conditions used to evaluate the performance of the Heat Recovery Unit are presented in Table 5.3. The nominal air inlet temperature was based on the average of the five manifold thermocouples.

Table 5.3. Thermal test matrix

Nominal Flow Rates		Nominal Air Inlet Temperature (°C)			
		300	400	450	500
Air Flow (g/s)	Oil Flow (g/s)	Warm Oil (73 °C Inlet)		Cold Oil (53 °C Inlet)	Warm Oil (73 °C Inlet)
20	40	x	x	x	x
	47	x	x	x	x
	55	x	x	x	x
23	40	x	x	x	x
	47	x	x	x	x
	55	x	x	x	x
27	40	x	x	x	Design
	47	x	x	x	x
	55	x	x	x	x

The motivation behind the selected test values was to operate the HRU in a range that encompassed the design air flow rate, oil flow rate, and air inlet temperature. However, equipment limitations altered the originally intended test values. The maximum air flow rate was dictated by the blower and the back pressure caused by the HRU. At the highest blower flow rate, the design air flow rate was achieved. The oil flow rates originally bracketed the design flow rate; however after calibrating the flow meter using a catch and weigh method (see Section 5.3.4.2), the actual flow rates were higher than the measured flow rate. The oil inlet temperature was tested off design due to a high system pressure during testing. For the inlet air temperature of 450 °C, two oil inlet temperatures were tested for comparison. The values in Table 5.3 reflect the experimental values tested.

For each thermal test case, steady state measurements were taken at 100 samples per second for three minutes. The thermocouple measuring the outlet air temperature from the HRU was located approximately in the center of the outlet plane and about 1 mm away from the face.

5.3.3 Traverse Measurements

The traverse system was designed to move the hot wire and thermocouple sensors in a plane parallel to the exposed face of the Heat Recovery Unit. Measurements were taken using a stop-and-go style of traversing. The hot wire and thermocouple were moved to a measurement position, then data was acquired for a specified period of time before moving on to the next point. Two movement sequences were developed: a column-wise traverse and a row-wise traverse, as shown in Figure 5.9.

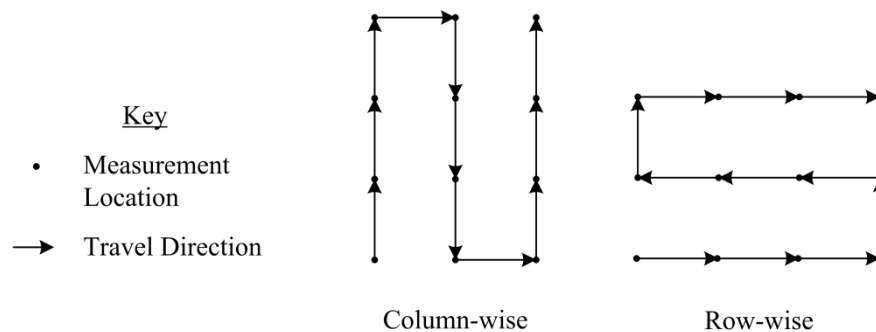


Figure 5.9. Scanning sequence styles

These two sequence styles were mainly developed for program flexibility. The coordinate system used for the traverse measurements is drawn in Figure 5.10. This coordinate system is only an indication of direction; it is not the origin of each scan. Each velocity scan has its own origin. The channels are numbered starting from the bottom left corner. The numbering

continues to ascend traveling right then up. Figure 5.11 shows the channel numbering and spacing of the air channels.

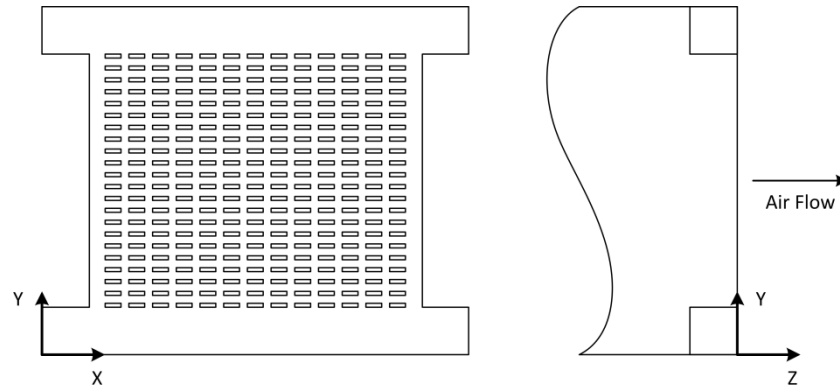


Figure 5.10. Coordinate system for the traverse measurements

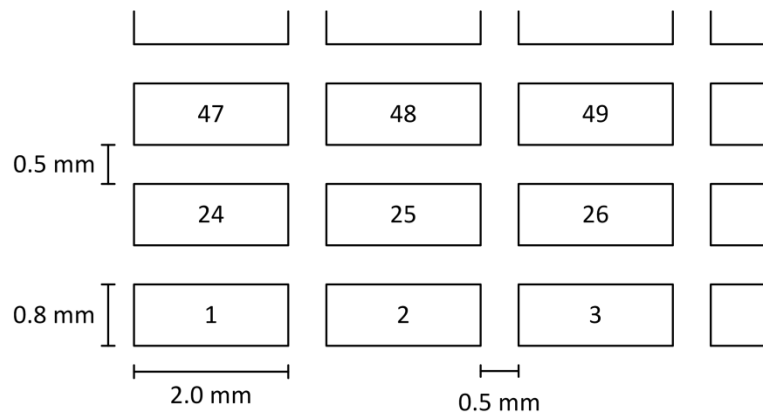


Figure 5.11. Air channel numbering and spacing

Velocity measurements were taken with unheated air at the design flow rate of 27 g/s through the Heat Recovery Unit and no oil flow. In order to get adequate resolution of the channel velocities, measurements were taken in several regions instead of the full field because the acquisition time of a full field profile would have been impractical. To understand general trends in the air flow, the hot wire probe was traversed horizontally and vertically

along the sides and middle channels of the HRU, as shown in Figure 5.12. Window regions were also scanned to examine the flow structures near the channel outlets.

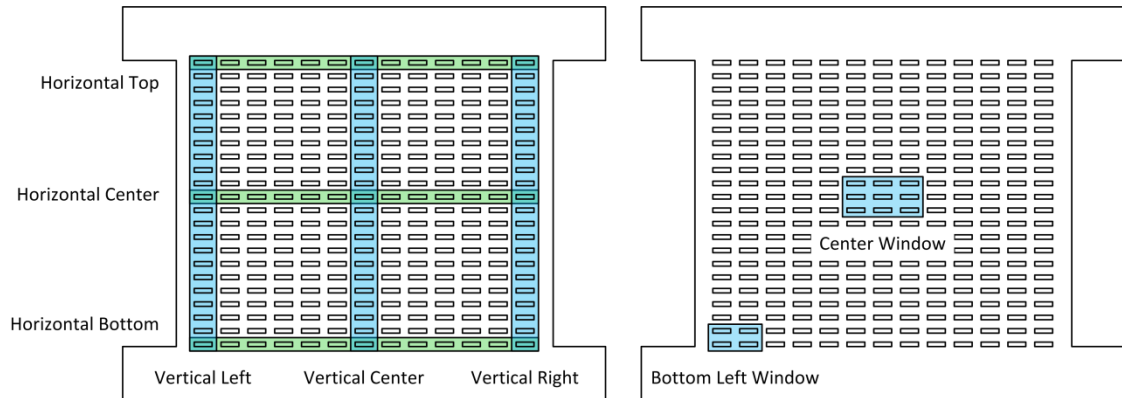


Figure 5.12. Hot wire measurement regions: vertical and horizontal regions shown on the left, window regions shown on the right

Vertical scans were executed using the column-wise movement scheme and the horizontal scans used the row-wise scheme. The window scans used the column-wise scheme. The hot wire probe was oriented such that the wire was perpendicular to the primary direction of travel. For the vertical and window scans, the wire was mounted horizontally. For the horizontal scans, the wire was mounted vertically. The scan resolution for each region is presented in Table 5.4.

The scan resolutions differ because of acquisition timings. The time it took to complete a scan (horizontal or vertical) was on the order of 3 to 3.5 hours. If a horizontal scan was held at the same resolution of a vertical scan, then the measurement acquisition time would have been approximately 2.5 hours. Increasing the resolution of the horizontal scans only added a little more time to the process because the travel distance in the x-direction of a horizontal was less than the travel distance in the y-direction of a vertical scan. This allowed the horizontal scans

to be more resolved in the y-direction, while not significantly changing the amount of time required to acquire the data at the same sampling rate.

Table 5.4. Scan resolutions

Scan Case	Directional Resolution	
	X (μm)	Y (μm)
Horizontal Top	200	100
Horizontal Center	200	100
Horizontal Bottom	200	100
Vertical Left	200	200
Vertical Center	200	200
Vertical Right	200	200
Center Window	100	100
Bottom Left Window	100	100
Full Temperature Field	1000	1000

For all hot wire scan cases, at least 1,000 samples were collected over a period of one to two seconds for each measurement location. All velocity measurements were taken approximately 1 mm away from the face of the Heat Recovery Unit. The justification for this distance comes from measurements of the velocity in the z-direction for a center exhaust channel. As can be seen from the graph in Figure 5.13, the velocity was nearly constant within the range of 0.5 mm to 5 mm, then it decreased as the probe moved away from the HRU face. This behavior resembles classical free jet theory.

Unfortunately, the hot wire anemometer broke before simultaneous velocity and temperature measurements could be taken for a thermally loaded condition. However, a full field temperature profile was captured for the nominal conditions of 27 g/s air flow, 450 °C air inlet temperature, 55 g/s oil flow, and 60 °C oil inlet temperature. The resolution for the thermal profile was 1 mm in both the x-direction and y-direction. Four hundred measurements

were taken over three seconds for each location. The thermocouple was located approximately 2 mm away from the face of the HRU.

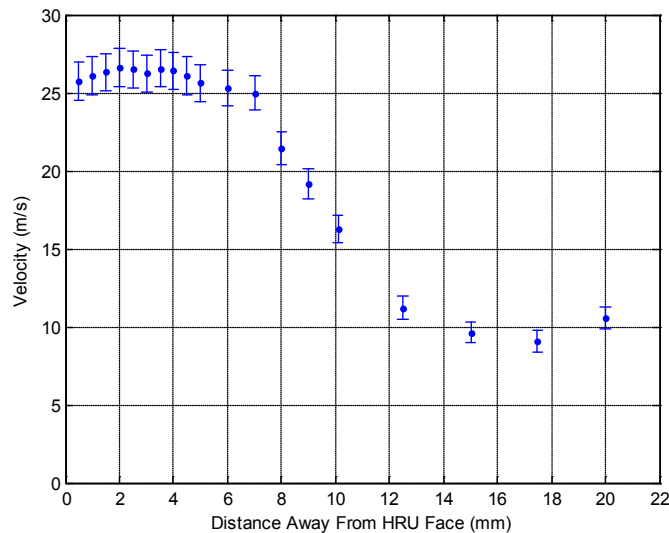


Figure 5.13. Velocity in the z-direction

Parallelism of the hot wire traverse to the face of the Heat Recovery Unit was not explicitly measured. Instead, the parallelism was visually inspected by placing the hot wire probe support, without the hot wire, such that it barely touched the face of the HRU. The hot wire was then traversed to the extreme locations of the channel array (i.e. the top-left, top-right, bottom-left, and bottom-right points) to verify parallelism.

5.3.4 Equipment Calibration

5.3.4.1 Hot Wire Anemometer Calibration

Calibration of the hot wire anemometer was accomplished with a TSI Model 1125 Calibrator. A special flow chamber on the Model 1125 Calibrator distributes the air evenly in

the chamber before it exits through an orifice plate. The air velocity at the exit of the orifice plate can be calculated using a form of Bernoulli's equation, shown in Equation (5.1).

$$v = \sqrt{\frac{2\Delta P}{\rho}} \quad (5.1)$$

The velocity depends on the differential pressure across the orifice and the air density. Inside the flow chamber, shown in Figure 5.14, the velocity was assumed to be zero. TSI verified the chamber velocity distribution was by laser Doppler velocimetry. An OMEGA PCL-1B pressure calibrator with a 0-10 in H₂O differential transducer was used to measure the chamber pressure during the calibration.

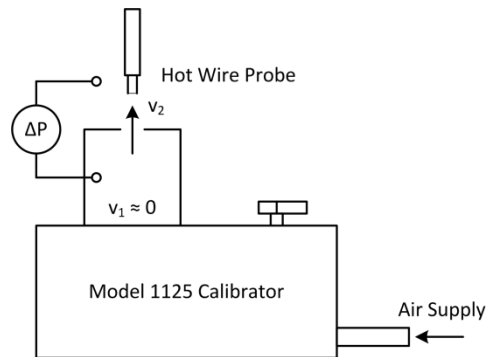


Figure 5.14. Hot wire calibration system

5.3.4.2 Flow Meter Calibration

The signal conditioners for the air and oil turbine flow meters were calibrated with a frequency generator and multimeter. A list of the equipment used to calibrate the signal conditioners is presented in Table 5.5. The mass flow rates calculated from measurements of volumetric flow from the oil flow meter were correlated to mass flow rates by a catch and weigh system for temperatures ranging from 50 °C to 100 °C. These tests were performed to verify the accuracy of the mass flow calculations when based on the flow meter

measurements. The oil catch and weigh system is drawn in Figure 5.15. The results from the catch and weigh indicated that the oil flow meter measured a volume flow rate that was less than the actual flow rate. This was also verified with an inline flow meter. The pressure transducers used in the oil catch and weigh system and in the thermal test bench were calibrated with an OMEGA DPG4000 pressure calibrator.

Table 5.5. Equipment used to calibrate the signal conditioners

Item	Manufacturer	Model	Serial
Function Generator	Tektronix	CFG280 11MHz Function Generator	CFG-280TW53825
Power Supply	Tektronix	CPS250 Triple Output Power Supply	CPS-250TW18981
Voltage/Current Multimeter	Fluke	45 Dual Display Multimeter	10960100
Frequency Reader	Fluke	175 True RMS Multimeter	7786013

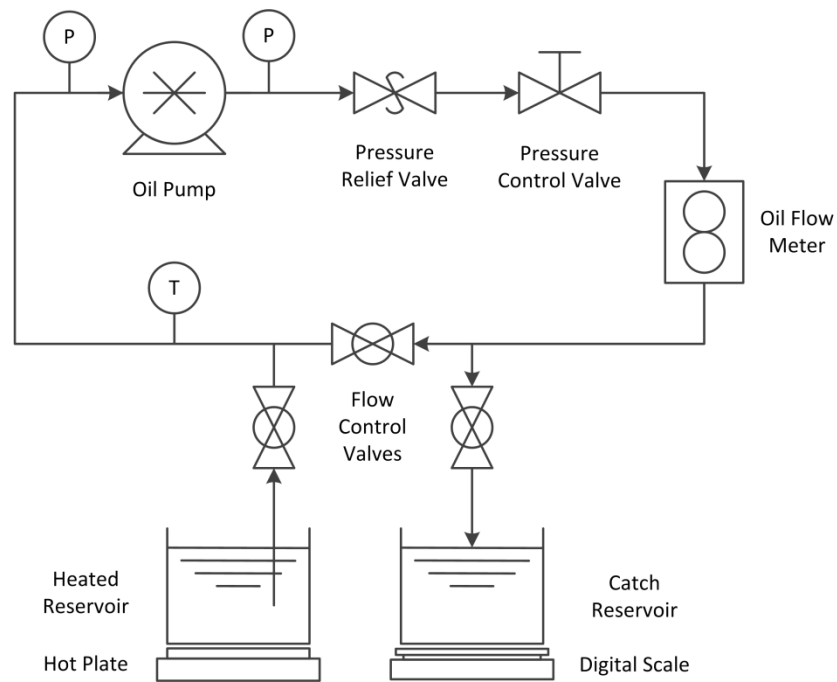


Figure 5.15. Oil flow meter calibration system

6 Data Analysis

In the first part of this chapter, the calculations and data reduction for the Microscope Image Analysis, Shim Profile Analysis, thermal tests, and traverse profiles are presented. The second part contains the uncertainty analysis for each process.

6.1 Data Reduction and Calculations

6.1.1 Microscope Image Analysis

The Microscope Image Analysis utilizes MATLAB's Image Processing Toolbox to compute the cross-sectional area, perimeter, and hydraulic diameter of the air channels on the bonded Heat Recovery Unit. The image processing uses edge detection and morphology tools to isolate individual channels and clean them before calculating the channel geometries.

Additionally, polynomial masks are used to enhance the definition of the microchannel shapes. Detailed steps of the process are outlined below.

1. Load the original image
2. Convert the colored image to grayscale
3. Adjust the contrast
4. Convert to a black and white (binary) image
5. Dilate the image
6. Detect the edges of the microchannels
7. Remove channels on the border
8. Remove small particles
9. Fill channel regions
10. Smoothen channels
11. Incorporate line and arc masks
12. Calculate channel areas
13. Calculate channel perimeters and hydraulic diameters

Suppose the microchannels in Figure 6.1 need to be measured. The first two steps are to convert the RGB image to grayscale then adjust the contrast.

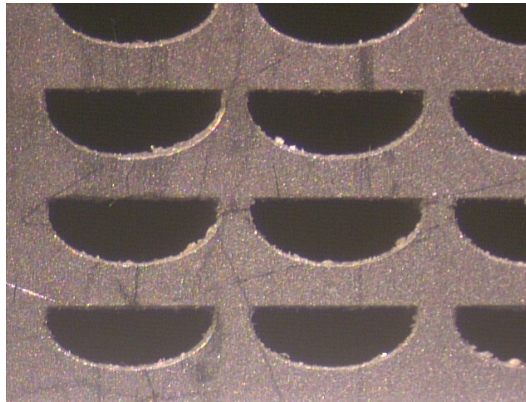


Figure 6.1. Example microscope image of the air channels

Adjusting the contrast makes it easier to distinguish the channel openings from the walls. The difference is noticeable between Figure 6.2 and Figure 6.3. The image is then converted into a black and white picture for morphological operations (Figure 6.4). The first operation on the black and white image is a dilation. A dilation expands the white pixels and removes most of the small particles from the image conversion from a grayscale to black and white. An image of the dilate result is shown in Figure 6.5.

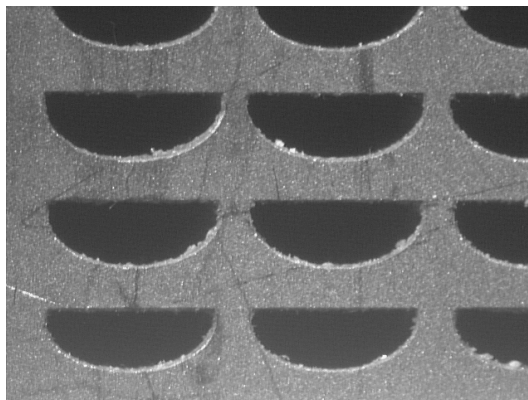


Figure 6.2. Convert image to grayscale

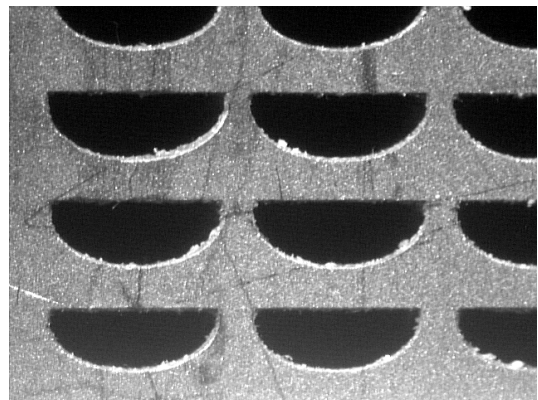


Figure 6.3. Adjust contrast

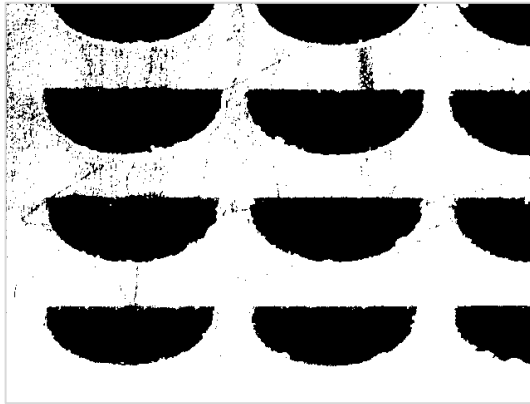


Figure 6.4. Convert image to black and white (binary)

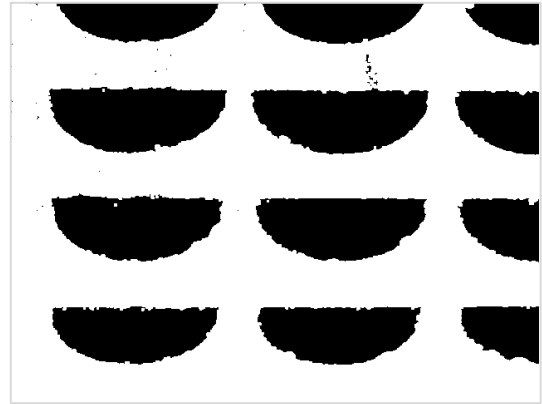


Figure 6.5. Dilate the image

Next, the channels are outlined using Canny edge detection, as shown in Figure 6.6. The incomplete channels are removed by clearing any white pixels that are connected to the border of the image. Only the complete channels and a few small particles remain in Figure 6.7.

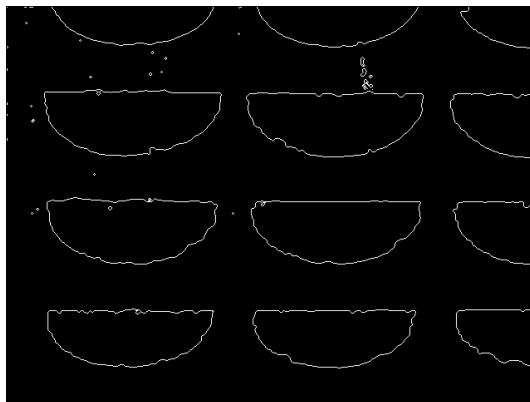


Figure 6.6. Channel edge detection

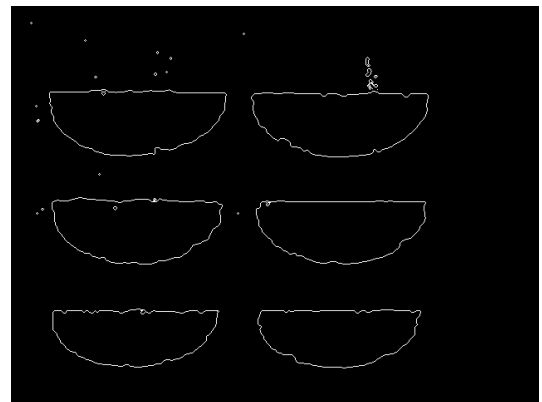


Figure 6.7. Incomplete channels removed

The small particles are removed by setting a threshold for the size of the regions that should remain in the image. Afterwards, the channels are filled in then smoothed with another dilate step.

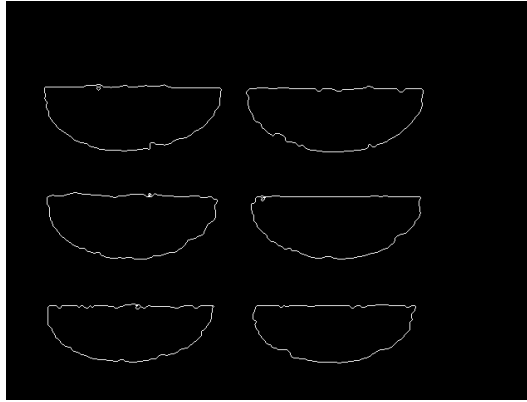


Figure 6.8. Small particles removed

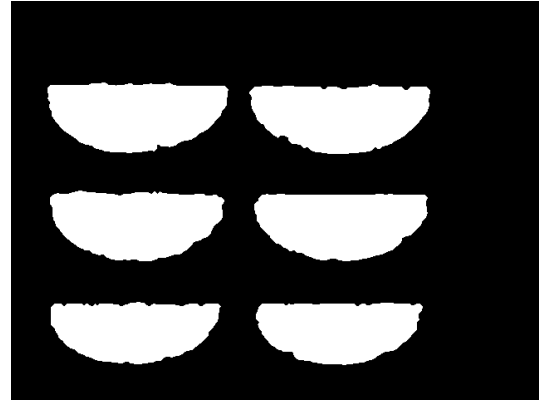


Figure 6.9. Filled and smoothened channels

A unique feature of this analysis is the use of line and arc masks to enhance the shape of the channels. A line is fitted to the top of a channel using the top-left and top-right points of the channel region. The remaining points on the periphery of the channel region are fitted with an 8-th order polynomial. This order was chosen because it was the lowest order that produced smooth and visually accurate results. The line and arc masks are illustrated in Figure 6.10. The channel regions are filled after the line and arc masks are merged with the image. Figure 6.11 shows the six example channels after the masks have been applied.

The channel areas are calculated by counting the pixels in each region, then applying a pixel to area conversion factor from the microscope calibration process. The final result of the area calculation is shown in Figure 6.12.

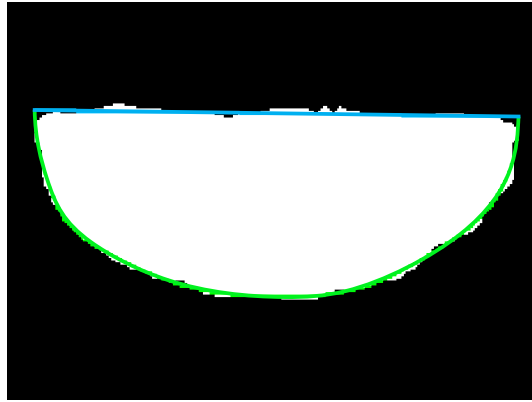


Figure 6.10. Line and arc masks on a channel

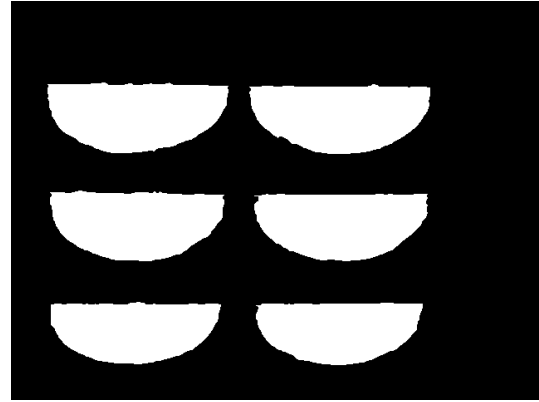


Figure 6.11. Masks incorporated and channels cleaned

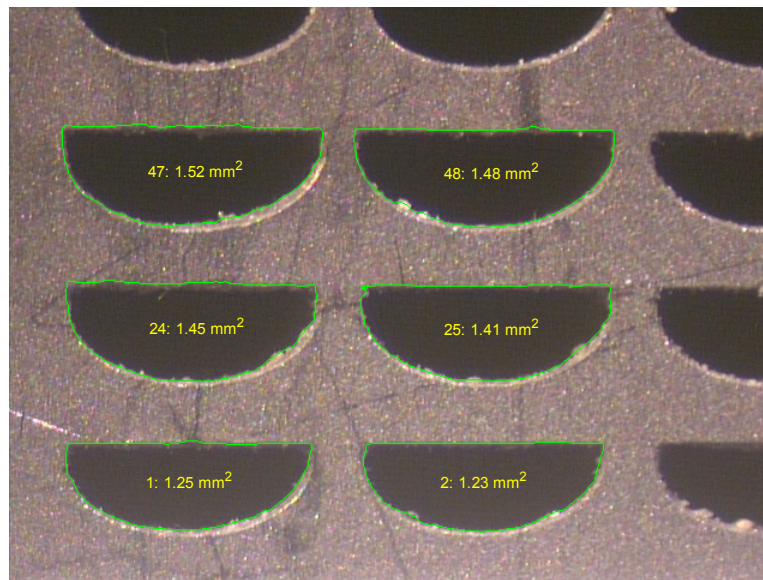


Figure 6.12. Example result of the Microscope Image Processing

The perimeter is estimated by the lengths of the line and arc masks. The arc length was calculated numerically from

$$L_{curve} = \int_a^b \sqrt{1 + \left(\frac{dy}{dx}\right)^2} dx \quad (6.1)$$

The derivative of the arc was approximated with central differencing. Forward and backward differencing were used at the ends.

$$\text{Central differencing: } \left. \frac{dy}{dx} \right|_i = \frac{y_{i+1} - y_{i-1}}{x_{i+1} - x_{i-1}} \quad (6.2)$$

$$\text{Forward differencing: } \left. \frac{dy}{dx} \right|_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \text{ for } i = 1 \quad (6.3)$$

$$\text{Backward differencing: } \left. \frac{dy}{dx} \right|_i = \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \text{ for } i = n \quad (6.4)$$

The integral was evaluated using the trapezoidal rule. Finally, the hydraulic diameter was computed as

$$D_h = \frac{4A}{P} \quad (6.5)$$

6.1.2 Shim Profile Analysis

The Shim Profile Analysis measured the cross-sectional areas, perimeters, and hydraulic diameters of the channels on the air and oil shims. Several different programs and files were used to collect and process shim data. Figure 6.13 shows the overall process of the profiling analysis.

The first step was to collect the three-dimensional shim profile data using the ZeScope. Second, SPIP was used to average the three-dimensional data into a two-dimensional profile of the channels. Next, there was a file conversion to store the profile data in a form that was quick and convenient for MATLAB to access during the calculation processes.

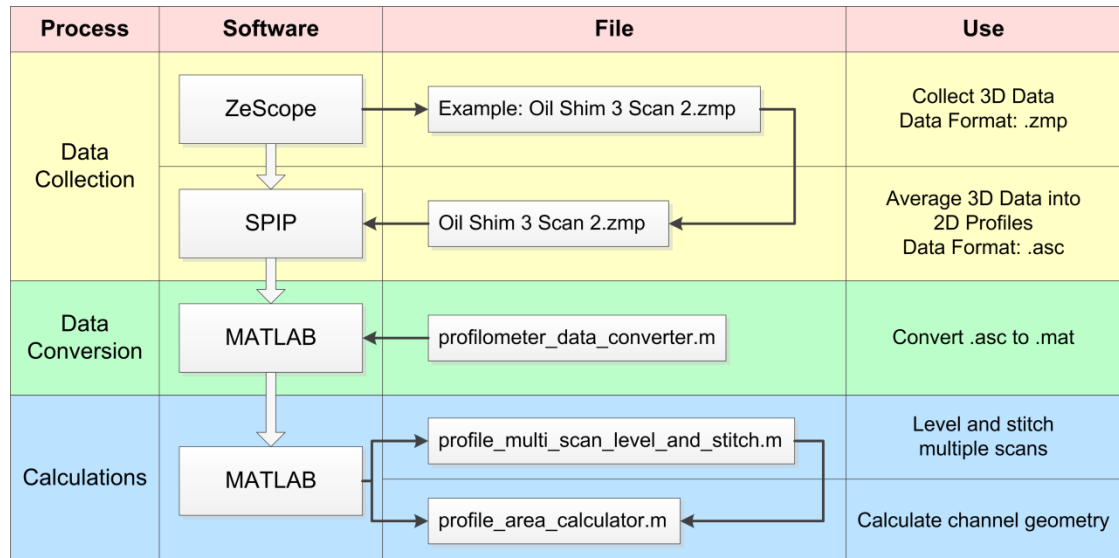


Figure 6.13. Profilometer processing diagram

Two MATLAB programs were developed to perform profile calculations. The first program is the Channel Geometry Calculator. This program was designed to perform all the necessary calculations for a single scan. A scan may contain one or more full channels to be processed. The second program that was developed is the Multi-scan Leveler and Stitcher. The Leveler and Stitcher combines multiple scans together for use with the Channel Geometry Calculator. The calculation processes for both programs are explained in the following sections.

6.1.2.1 Channel Geometry Calculator

The process of calculating the channel geometries is listed below.

1. Load data
2. Convert units if necessary
3. Calculate the first derivative via central differencing
4. Identify values where the derivative is near zero
5. Separate shim tops and bottoms
6. Determine edges of shim tops and the number of channels present
7. Draw lines at the shim tops using a line fit
8. Draw lines over the tops of channels to estimate the channel area
9. Calculate channel areas using integration
10. Calculate channel perimeters and hydraulic diameters

The first step is to load the profile data. An example of profile data for one scan of an oil shim is shown in Figure 6.14. Note that the scales on the axes are not the same, thus features such as the curvature on the bottom of the channels may be exaggerated.

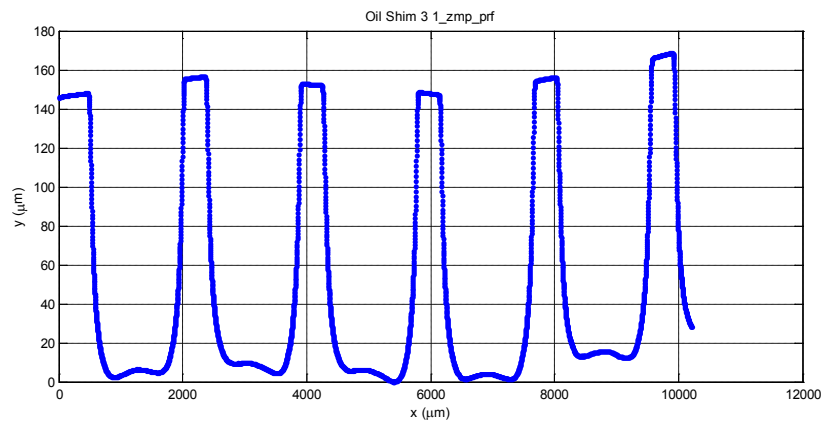


Figure 6.14. Example profile data

In step 3, the first derivative is numerically computed via central differencing. An example result is displayed in Figure 6.15. The first derivative is used to distinguish areas that are the tops of the shims and bottoms of the channels.

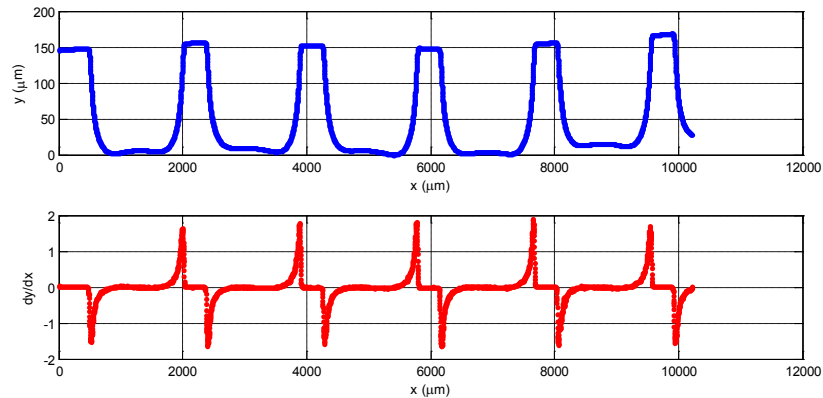


Figure 6.15. Profile data and the first derivative

Tops and bottoms occur where the derivative is near zero. Values that are considered to be near zero lie within a specified percentage of the maximum absolute value of the first derivative. For example, if the maximum absolute value is 100 and the tolerance is set to 1%, then values between -1 and 1 would be considered sufficiently close to zero. For the data in Figure 6.16, the zero tolerance was set to 1%. The results are plotted in green for both the first derivative and the original profile data.

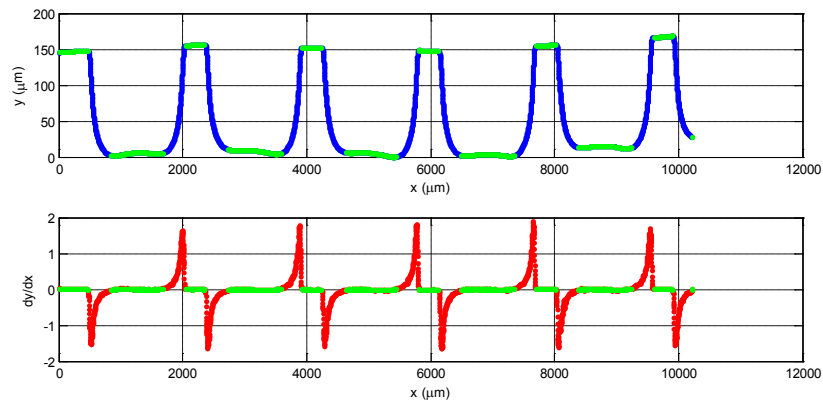


Figure 6.16. Identify derivatives near zero

Separating the tops of the shims from the bottoms of the channels was accomplished by drawing a horizontal line in the x-y data. The location of the line was based on a user selected percentage of the maximum y value. In general, if the profile has a large bow, then the horizontal line should be lower (smaller y value) in order to sufficiently separate the two sets. For the results in Figure 6.17, the separation line was at 60% of the maximum y value.

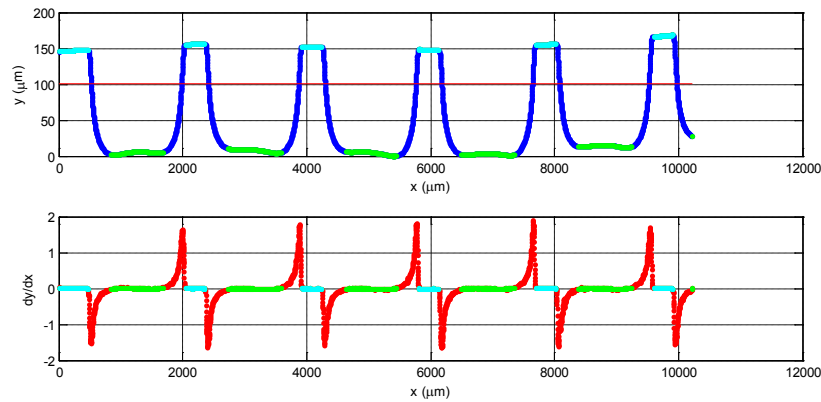


Figure 6.17. Tops (cyan) and bottoms (green) separated by the horizontal line in the x-y data

Once the tops and bottoms have been identified, the next step is to separate the tops from each other because at this point they are all lumped together in the same vector. To separate the tops, the change in x for all the top values is calculated. When the change in x is large, indicating the gap of the channel, the tops should be separated. The change in x is defined as

$$\Delta x_i = 0 \text{ for } i = 1$$

$$\Delta x_i = x_{tops}|_{i+1} - x_{tops}|_i \text{ for } i > 1 \quad (6.6)$$

As seen from Figure 6.18, it is fairly easy to identify where there is a gap between the tops due to the presence of a channel. Using the same method of distinguishing shim tops and channel

bottoms, a horizontal line is drawn to separate the large Δx values. The number of channels can be computed by noting the number of large discontinuities in x .

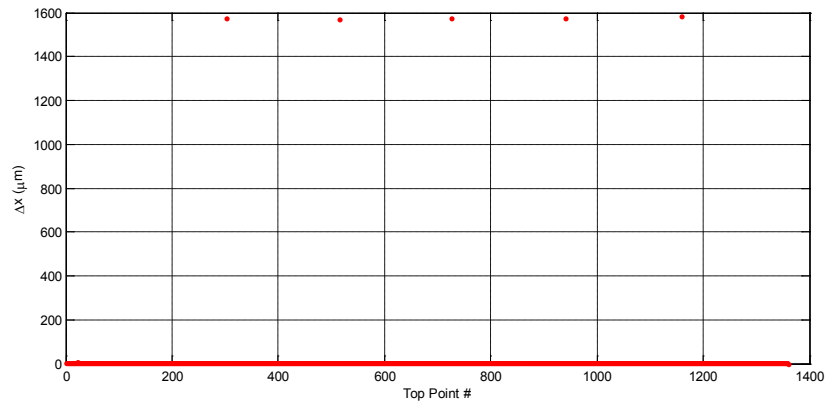


Figure 6.18. Change in x for top points

By knowing the places where Δx is large, the top edges of the shims can be located. The edges are key points in the processing because they indicate where channels exist. These points are marked as red dots in Figure 6.19.

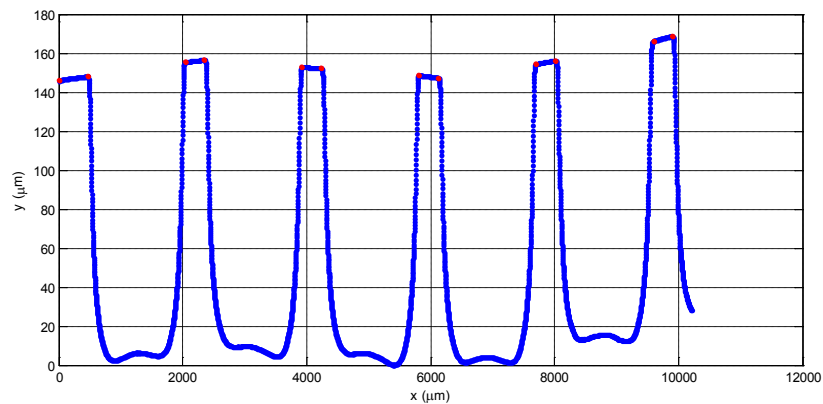


Figure 6.19. Edges of the shim tops marked in red

Channel areas are found by first forming a line on each of the shim tops. One of the fitted lines is plotted in cyan in Figure 6.20. Next, a line is drawn across the channel opening using the end points of the shim top lines. Two ends of a channel top line are plotted in Figure 6.21. All the channel tops for the example data are shown in Figure 6.22.

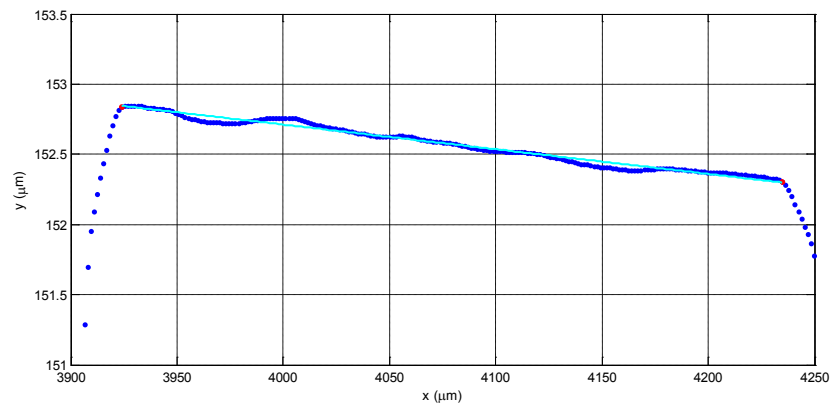


Figure 6.20. Line fit on shim top

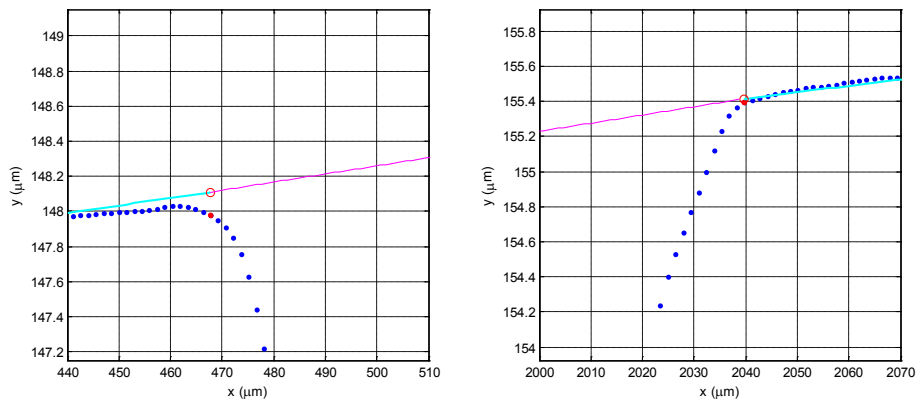


Figure 6.21. Left and right ends of a channel top

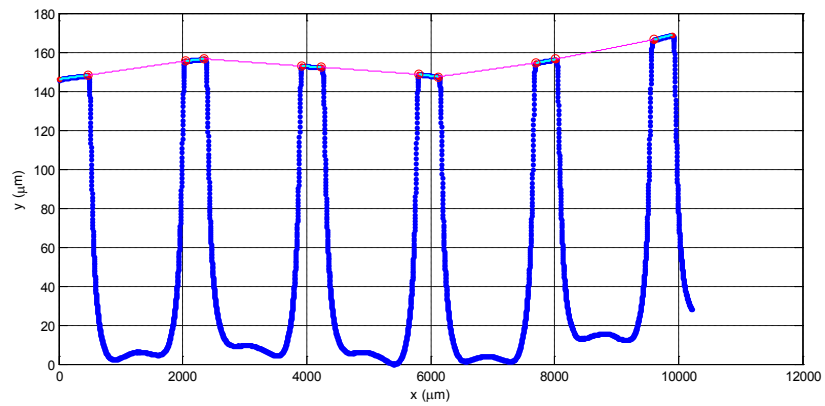


Figure 6.22. Channel tops

The channel areas are calculated by integrating the channel top line and subtracting the integral of the channel opening, or profile. This process is highlighted in Figure 6.23 through Figure 6.25. The trapezoidal rule was used for numerical integration.

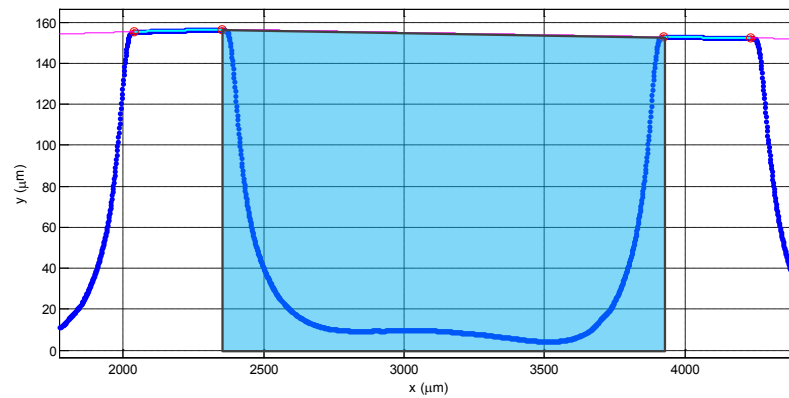


Figure 6.23. Integration area of the channel top line

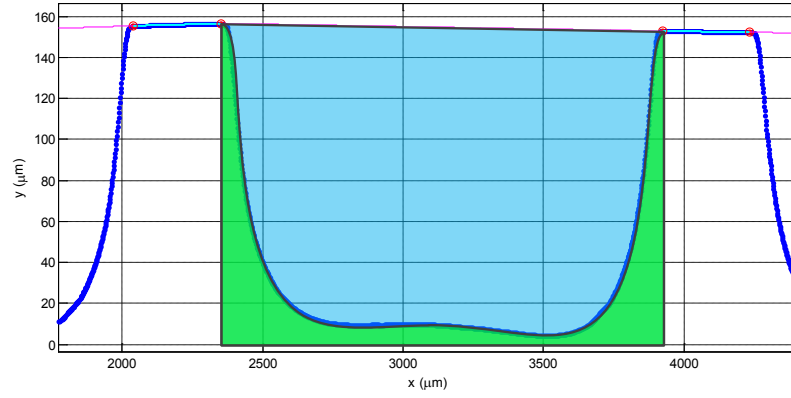


Figure 6.24. Integration areas of the channel top line and the channel surface

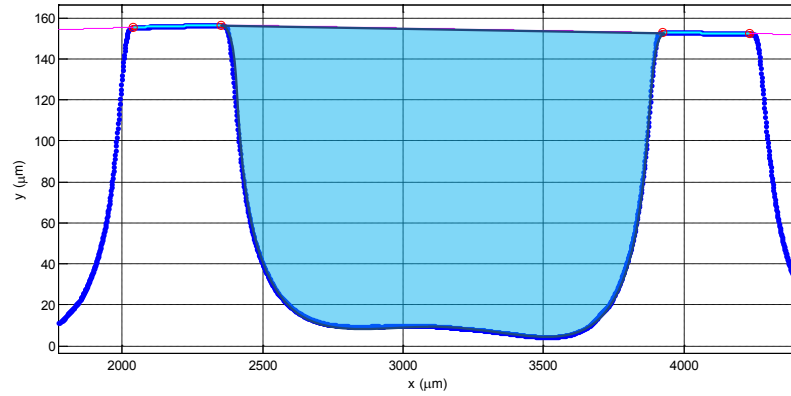


Figure 6.25. Channel area

A channel perimeter is the sum of the lengths of the channel top and the channel curve. The length of the top channel is simply calculated using the distance formula between two points. The length of the channel curve is found by calculating the arc length, as in Equation (6.1).

$$L_{curve} = \int_a^b \sqrt{1 + \left(\frac{dy}{dx}\right)^2} dx \quad (6.1)$$

The first derivative is already known from previous calculations. The arc length integral is computed using the trapezoidal rule. Finally, the hydraulic diameter is calculated as

$$D_h = \frac{4A}{P} \quad (6.5)$$

6.1.2.2 Multi-scan Leveler and Stitcher

The Multi-scan Leveler and Stitcher served two purposes. First, the profilers were leveled such that the shim tops were flat, as they would appear after bonding. Second, the stitching process combined multiple scans together such that all the channels of the same shim could be directly compared and processed. The calculation procedure is listed below. Steps 1 through 8 of the leveling and stitching process are exactly the same as for the Channel Geometry Calculator.

1. Load data
2. Convert units if necessary
3. Calculate the first derivative via central differencing
4. Identify values where the derivative is near zero
5. Separate shim tops and bottoms
6. Determine edges of shim tops and the number of channels present
7. Draw lines at the shim tops using a line fit
8. Draw lines over the tops of channels
9. Extend top shim lines to the ends of the domain if necessary
10. Level the data using the fitted lines
11. Bring in the next scan and trim the edges
12. Shift the next scan and level it with the previous scan
13. Merge the overlapping area

Suppose the data in Figure 6.26 needs to be leveled. From the previous analysis (steps 1 through 8), the channel top lines and shim top lines can be identified for the given profile. For the left and right ends of the domain where the channels are incomplete, the shim top lines are

extended. These lines are used to level the data. For this analysis, all of the data is leveled with the maximum y value. Figure 6.27 through Figure 6.29 display the lines used for leveling.

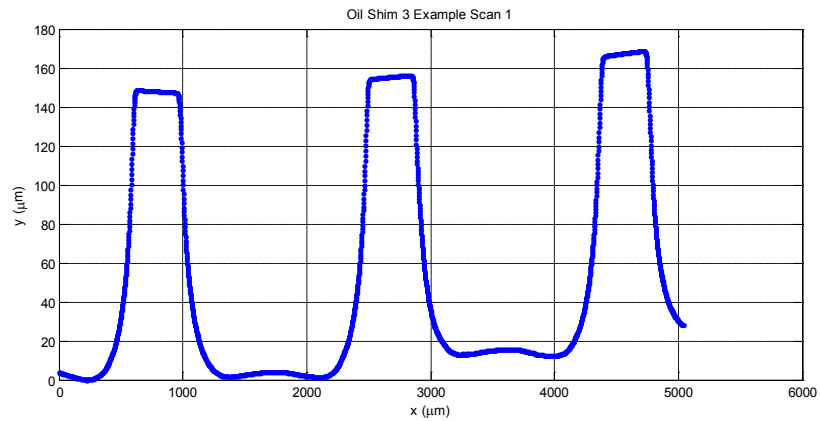


Figure 6.26. Example data to level

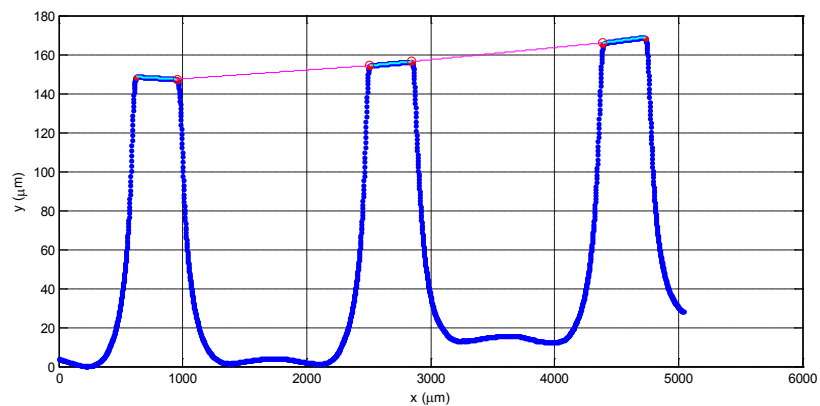


Figure 6.27. Channel lines and shim top lines drawn using the methods previously developed

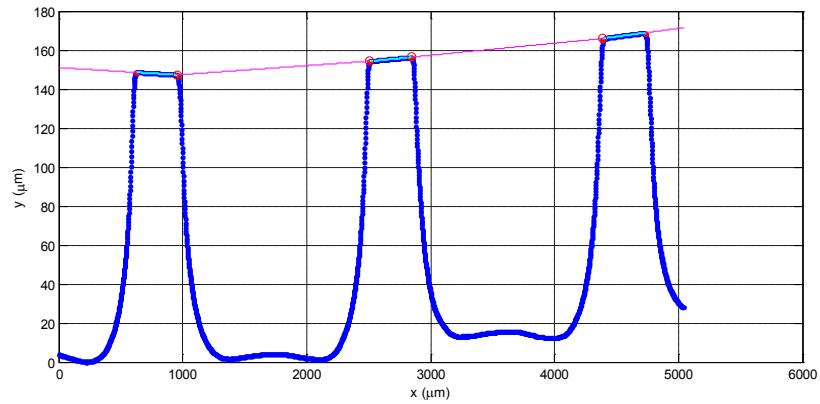


Figure 6.28. Shim top lines extended for the left and right ends

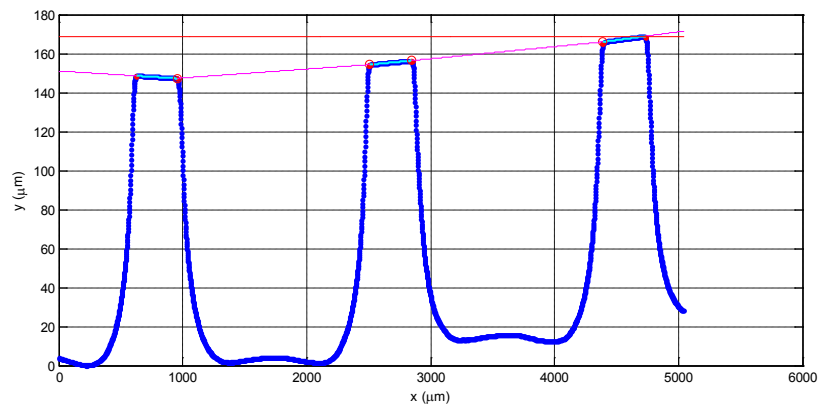


Figure 6.29. Level line at maximum y value

Each data point is shifted in height by the difference between its corresponding line value and the maximum y value. The result of this shift is shown in Figure 6.30.

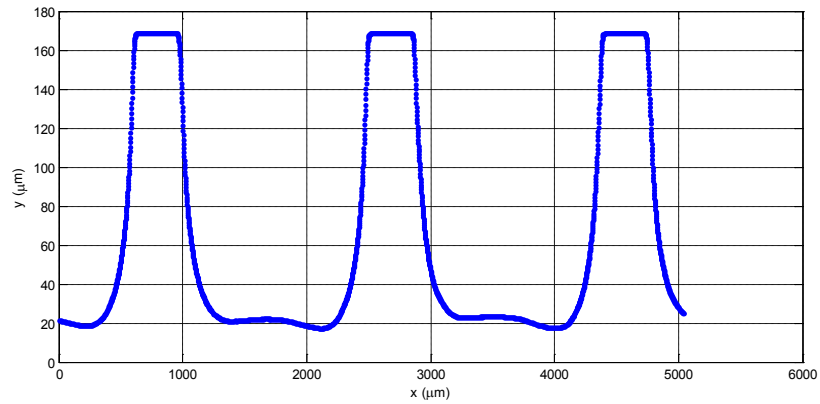


Figure 6.30. Leveled data

Suppose the two leveled scans in Figure 6.31 can be stitched together. The first step of the stitching process is to trim the scans at the ends. This is necessary because the merging points need to be comparable. In this analysis, it is assumed that the rightmost shim top of Scan 1 and the leftmost shim top of Scan 2 are physically the same. Therefore, the scans are trimmed up to the shim top edge points, as in Figure 6.32.

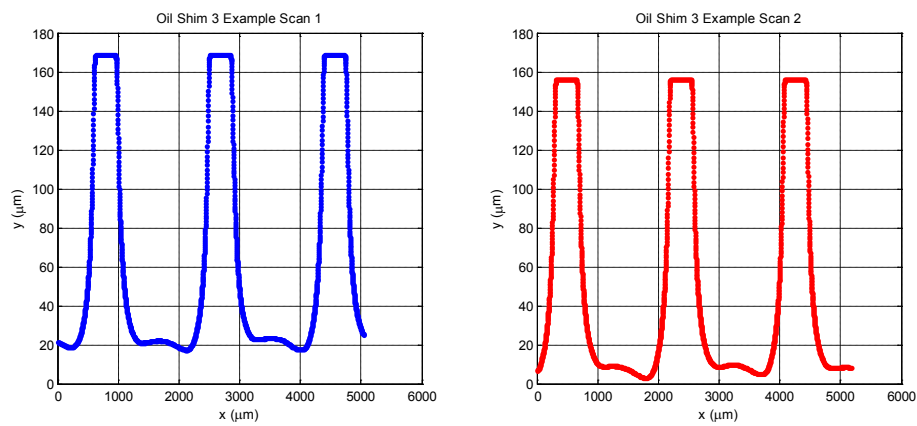


Figure 6.31. Two leveled scans to stitch

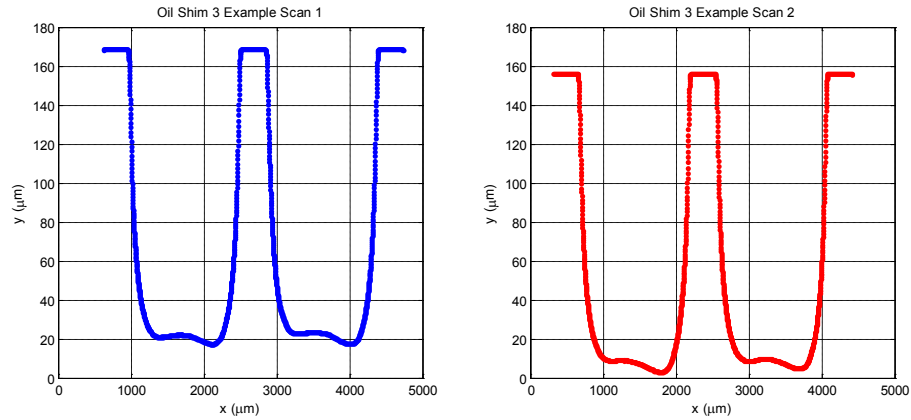


Figure 6.32. Trimmed scans

Figure 6.33 shows what the two scans look like when plotted on the same axis. Scan 2 needs to be adjusted so the twice-measured shim top is in the proper position. All the x values of Scan 2 are moved to the right such that the left edge of the shim top is in its corresponding position of Scan 1. Next, Scan 2 is leveled with Scan 1 by shifting all the y values of Scan 2. Figure 6.34 through Figure 6.35 illustrate the shifting process.

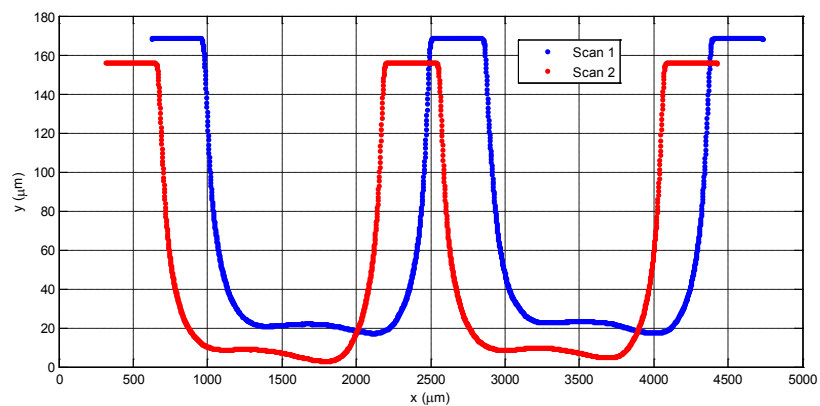


Figure 6.33. Both scans plotted together

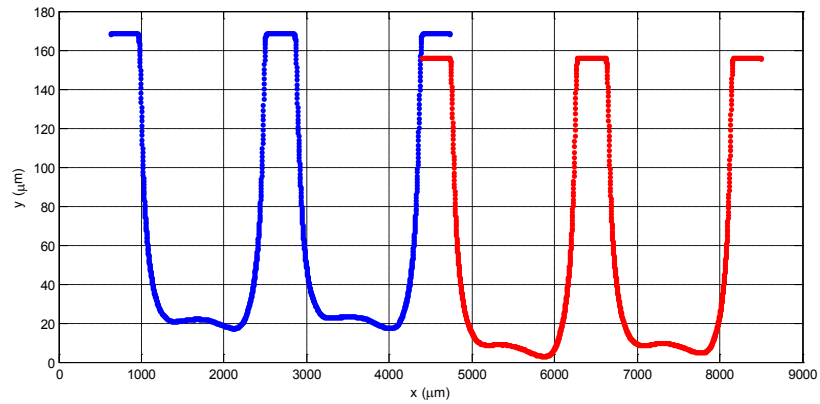


Figure 6.34. Scan 2 shifted right

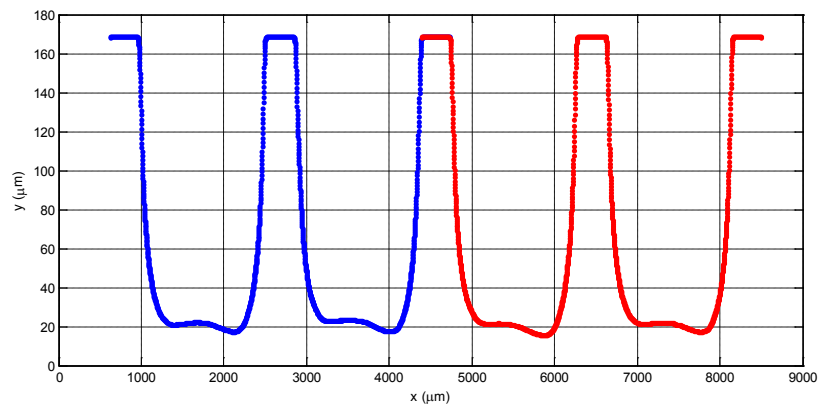


Figure 6.35. Scan 2 leveled with Scan 1

To merge the two data sets, the overlap region shown in Figure 6.36 needs to be examined. Merging the data is accomplished by averaging the x and y values of both data sets. Since it is unlikely that the overlap region contains the same number points in each scan, some points are not used in the averaging. The number of points taken for the averaging is the smaller of the number of points in each shim top. The averaging starts on the left, where Scan 2 was shifted to meet Scan 1. Figure 6.37 shows zoomed images of the left and right ends of

the overlap region. The end points of each scan's shim top are marked with a circle, and the points used in the averaging are connected with a solid line.

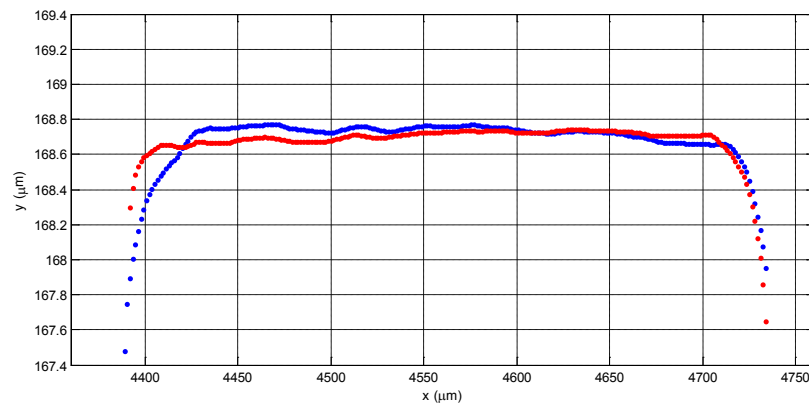


Figure 6.36. Overlap region

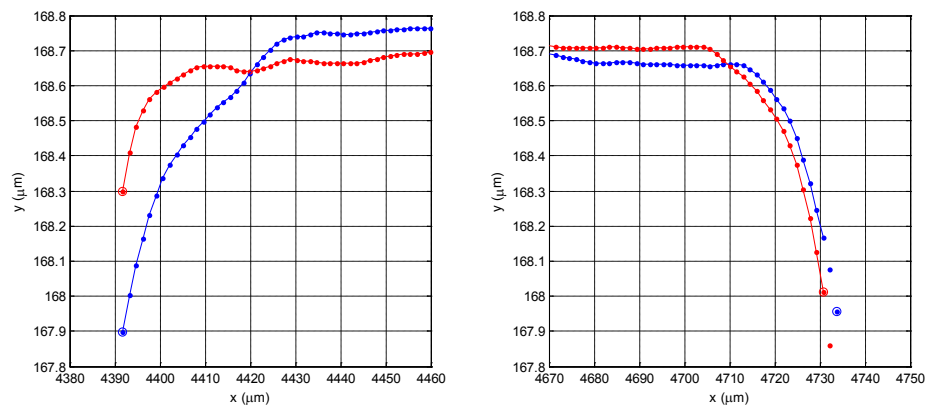


Figure 6.37. Left and right ends of the overlap region with end points circled and points for merging connected by a line

The merge points for the example overlap region are shown in Figure 6.38, and the final results of the stitching process are shown in Figure 6.39 and Figure 6.40.

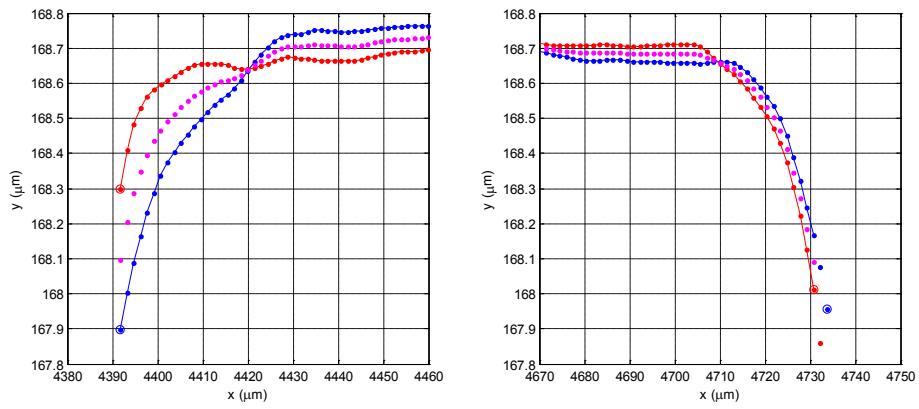


Figure 6.38. Calculated merge points (magenta)

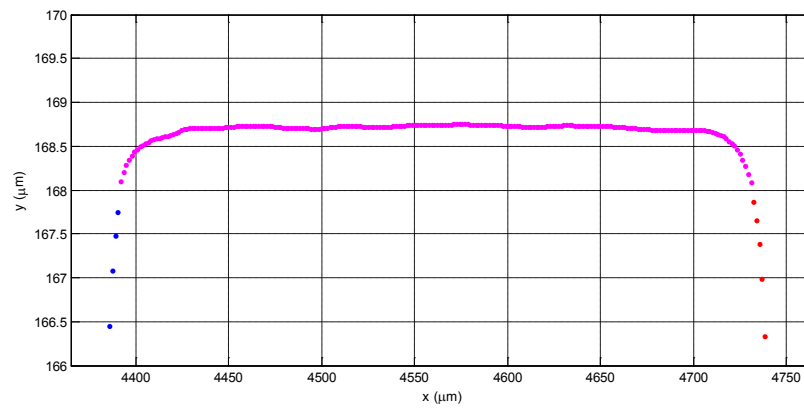


Figure 6.39. Merge result

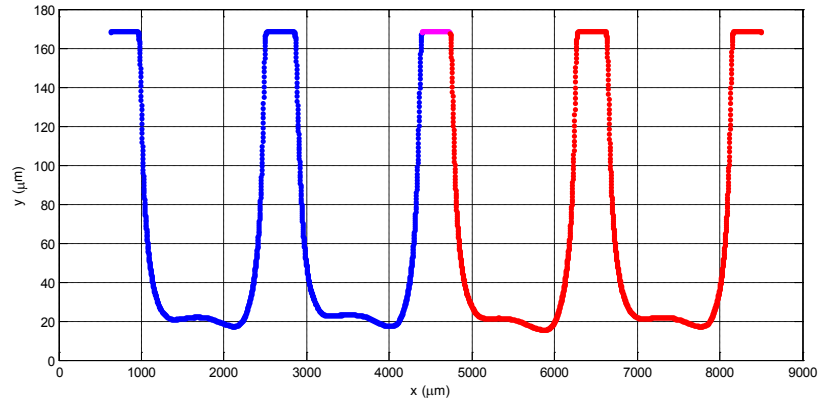


Figure 6.40. Stitch complete

With the completely stitched scans, the data was saved, then loaded into the Geometry Calculator for geometric measurements. There was no significant difference in the calculated area, perimeter, and hydraulic diameter between the leveled and unleveled profiles (less than 0.1% change).

6.1.3 Thermal Performance

The thermal performance of the Heat Recovery Unit is characterized by the duty, effectiveness, air pressure drop, and oil pressure drop. The duty is the amount of thermal energy that moves from one fluid to the other. Theoretically, the energy lost by the air stream should be picked up by the oil stream. Equation (6.7) shows that the thermal energy of a fluid is a function of the mass flow rate, the specific heat, and the temperature difference between the hot and cold temperatures.

$$\dot{Q} = \dot{m}c(T_h - T_c) \quad (6.7)$$

$$\dot{m} = \rho\dot{V} \quad (6.8)$$

The mass flow rate is the product of the density and the measured volumetric flow rate. Curve fits were produced for the density and specific heat of Paratherm NF oil, based on the manufacturer's provided data. For air, the density and specific heat were taken from Engineering Equation Solver (EES) [56]. The oil density was evaluated at the temperature after the flat plate heat exchanger (refer to Figure 5.6 in Section 5.3.1) because it was the closest measurement to the oil flow meter. Similarly, the inlet ambient air temperature was used to calculate the density because it was the closest temperature measurement to the air flow meter. To evaluate the specific heat for both fluids, the averages of the hot and cold temperatures of the Heat Recovery Unit were used. For oil stream, the hot and cold temperatures were the outlet oil temperature and inlet oil temperature of the HRU, respectively. For the air stream, the hot temperature was the average of the five manifold thermocouple readings. The cold air temperature was taken from the thermocouple located in the center of the air exit plane. Using these air and oil temperatures, the effectiveness of the HRU is defined as

$$\varepsilon = \frac{T_{Air,h} - T_{Air,c}}{T_{Air,h} - T_{Oil,c}} \quad (6.9)$$

It is important to recognize that the fluid energies and the effectiveness were calculated using point measurements or an average of several point measurements in the case of the inlet air temperature. Theoretically, the bulk mean temperatures should be used in the calculations. Measuring the bulk fluid temperatures is a difficult task. For the inlet air temperature and both oil temperatures, the physical space for measurements was very limited. Introducing more thermocouples would have further disturbed the flow. For the outlet air temperature, a single

point measurement was used for all the thermal tests. An ideal measurement would have been the entire exhaust temperature profile for each thermal test case. However, this would have been very time consuming. So, in the interest of time, point measurements were favored over full temperature fields for each test. A full temperature profile was taken for one thermal case.

The oil pressure drop across the Heat Recovery Unit was taken as the difference between the two pressure measurements on the inlet and outlet of the oil line. The air pressure drop was measured by the single pressure transducer in the manifold.

6.1.4 Velocity and Temperature Profiles

The theoretical air velocity of a single channel of the Heat Recovery Unit, assuming that the channels are the same size and uniform flow is

$$V = \frac{\dot{m}}{\rho(A_c \times N)} \quad (6.10)$$

where \dot{m} is the air mass flow rate, ρ is the density of air, A_c is the cross-sectional area of an air channel, and N is the total number of air channels. The Reynolds number is defined as

$$\text{Re} = \frac{\rho V D_h}{\mu} \quad (6.11)$$

where ρ is the density, V is the velocity, D_h is the hydraulic diameter, and μ is the dynamic viscosity. The Reynolds number can also be written in terms of the mass flow rate

$$\text{Re} = \frac{\dot{m} D_h}{(A_c \times N) \mu} \quad (6.12)$$

The hydrodynamic entrance length of the channels can be calculated as

$$x_{fd,h} = 0.05 \text{Re} D_h \quad (6.13)$$

This quantity is the channel length required to reach fully developed flow. The entry length will be considered in the results of the air and oil pressure drops.

The velocity and temperature profiles were smoothed using a spatial averaging scheme. Each point is averaged with neighboring points up to two spaces away, including diagonals.

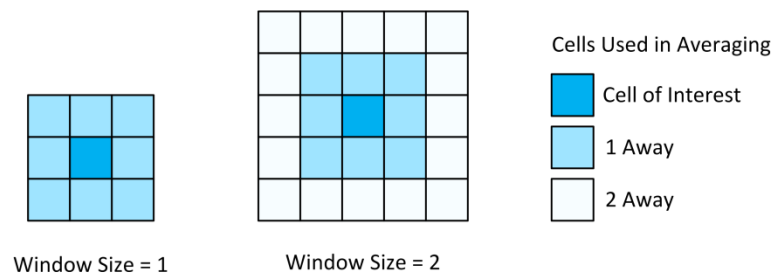


Figure 6.41. Averaging scheme for the velocity and temperature profiles

Since a full velocity profile was not obtained through measurements, a profile of the maximum velocity of all the air channels was calculated by interpolating the three horizontal and three vertical velocity scans. This calculation was performed to help visualize the relationships between the scans and to interpret the flow.

The first step toward creating the interpolated profile was to associate a maximum velocity for each of the measured channels. The horizontal and vertical scan data were collapsed by retaining only the maximum value along the minor dimension (i.e. collapse the y-direction for horizontal scans and the x-direction for vertical scans). Figure 6.42 shows the result of a collapsing operation for one of the horizontal scans.

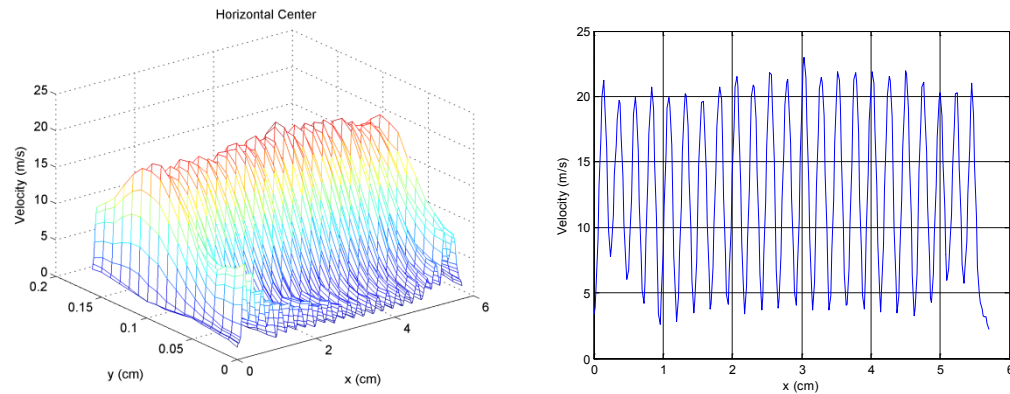


Figure 6.42. 3-D plot of the horizontal center velocity scan (left) and a plot of the collapsed data (right)

The maximum velocities occur where the spatial first derivative of the collapsed data crosses zero with a negative slope. Since this separation criterion is derivative based, the process may also detect small bumps in the ends of the data sets (see Figure 6.43). Only the first 23 points were taken for the horizontal scans and the first 45 of the vertical scans. There were some missing channels in the vertical direction; therefore only 45 channels were modeled, instead of the full 46 of the design. Explanations for what could have caused the shortage of channels in the vertical direction are discussed in Section 7.3.2. The extracted maximum channel velocities for the horizontal scan are shown in Figure 6.44. The uncertainty propagation for the velocity measurements is presented in Section 6.2.3 and discussed in Section 6.2.4.

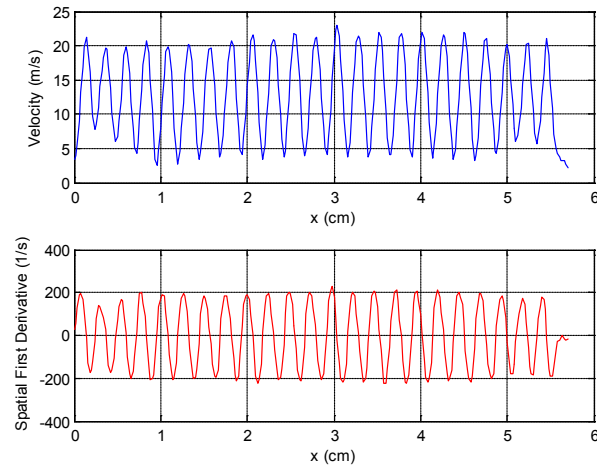


Figure 6.43. First derivative of collapsed data

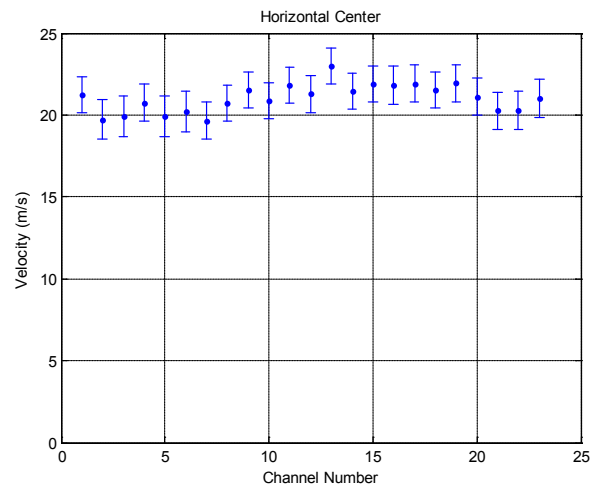


Figure 6.44. Maximum channel velocities for the horizontal center scan

A comparison of the maximum channel velocities for each scan revealed that the velocities did not match where the horizontal and vertical scans intersect (i.e. where the channels have been measured twice). The maximum channel velocities in the vertical scans were less than those of the horizontal scans. This was likely due to the different traversing

resolutions. The horizontal scans have a spatial resolution of 200 μm in the x-direction and 100 μm in the y-direction; whereas the vertical scans have a spatial resolution of 200 μm in both directions. Because the spatial resolution of the horizontal scans was higher than the vertical scans, the hot wire was able to resolve velocity profile of each channel better than the vertical scans. Therefore, the velocity profiles from the horizontal scans were believed to be more accurate. This is an important assumption in the velocity interpolation process.

Another factor that may have contributed to the difference in measured velocity is the orientation of the hot wire during the scans. For the horizontal scans, the sensor was positioned vertically, while for the vertical scans, the sensor was positioned horizontally. Recall that the hot wire sensing diameter was 25.4 μm and 0.25 mm long, and that the air channels were designed to be 2.0 mm x 0.8 mm. Theoretically, the orientation of the sensor should not affect the measurement as long as the sensor is placed perpendicular to the flow. A comparison between the horizontal scans and the window scans showed that the traverse resolution affected the velocity measurements more than the hot wire orientation. The window scans had a higher spatial resolution than the horizontal scans; 100 μm in both directions as compared to 200 μm x 100 μm . For the window scans, the hot wire was placed in a horizontal orientation and the traverse scheme was column-wise. The resulting velocity profiles showed that maximum measured velocities were between 20 and 25 m/s. This was consistent with the maximum velocities from the horizontal scans. The maximum velocities for the vertical scans were around 15 to 16 m/s (see Section 7.3.2 for data plots). Therefore, the difference in resolution of the horizontal and vertical scans caused an inconsistency in the measured maximum channel velocities.

To combine the horizontal and vertical maximum channel velocities for the interpolation process, the vertical channel velocities were shifted based on the difference between the velocities of the intersecting points (refer to Figure 6.45 and Figure 6.46).

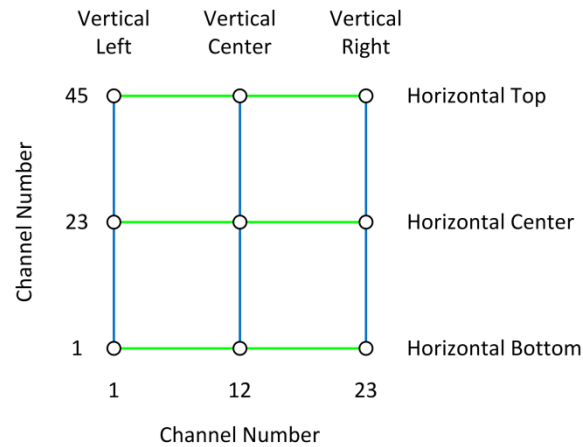


Figure 6.45. Intersecting points of the horizontal and vertical velocity scans

Each vertical scan shared a common channel with each horizontal scan. The differences in the maximum velocities for these channels were used to create shift lines, as shown in Figure 6.46. By adding the shift lines to the vertical scans, the channels common to both scan methods will have exactly the same velocity. For example, after the shift line is added, channels 1, 23, and 45 of the vertical left scan will have the same velocities as channel 1 from the horizontal bottom, center, and top scans, respectively. Once the shifts have been applied, the horizontal and vertical scans are combined into an array and put into a solver that approximates a partial differential equation to determine the maximum velocities of the unmeasured channels [57]. The results of the interpolation were smoothed with the nearest neighboring points. Extracted maximum velocities for each scan are provided in Appendix B. The interpolation results are presented in 7.3.2.3.

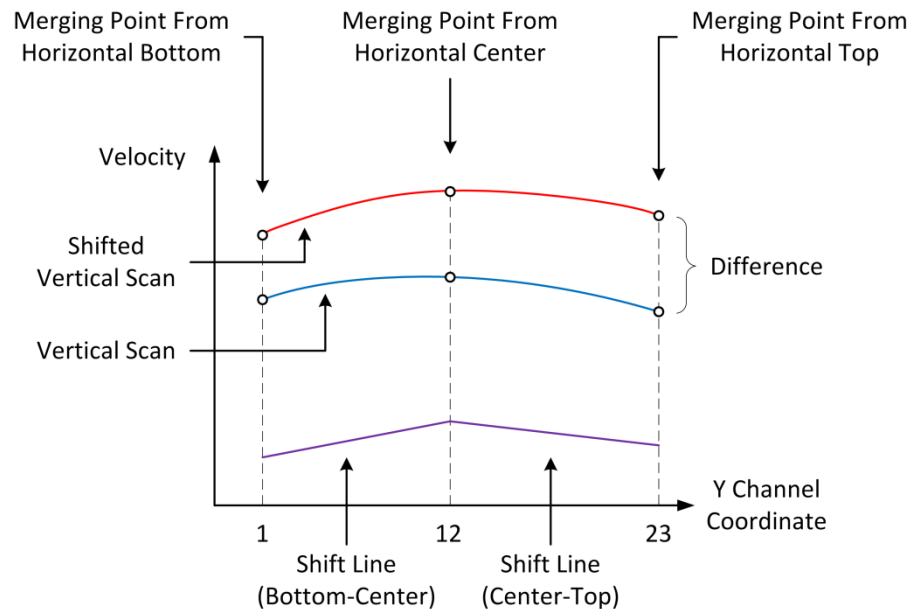


Figure 6.46. An illustration of the points used for calculating the shift lines with respect to a vertical scan

6.2 Uncertainty Analysis

6.2.1 Uncertainties in Microscope Image Processing

Uncertainties in the Microscope Image Processing primarily came from two places: the calibration and the image processing. In the calibration process, the pixel to length conversions can be altered by the placement of the VIA-100 measurement overlay on the markings of the calibration standard. The overlay had a limited resolution. At a magnification of 10x, the resolution was 10 μm per pixel in the x-direction and 10 μm per pixel in the y-direction. If the overlay was off by the width of the overlay line (two pixels) in both directions for the x and y calibrations, then the conversion factor in either direction changed by less than 0.1%, and the area factor changed by approximately 0.2 %. Since the changes were minimal,

the calibration error was considered negligible and was not included in the uncertainty calculations.

A more significant contributor to the uncertainty was the image processing method. The main goal of the image processing analysis is to draw an outline around the channels so the areas, perimeters, and hydraulic diameters can be calculated. The accuracy of the outline is critical to the calculations. Debris in the channels can have the most impact on the formation of the outlines and, therefore, the results. Much of the automated image processing is dedicated to cleaning the channels to minimize the effect of debris and to improve the channel shape. To estimate the accuracy of the automated outlining process, some channels were manually outlined then put into a simplified image processing program to calculate the areas. The example image used in the explanation of the image processing was also used for the manual outlining and uncertainty estimation because it had a variety of apparent channel sizes, and there was some debris in the channels. Manual outlining was done with GNU Image Manipulation Program (GIMP) 2.6. Figure 6.47 shows a side-by-side comparison of the results from the automated outlining process and the manual outlining process. The results are also tabulated in Table 6.1.

From the results of the outlining comparison, a conservative estimate of 6% uncertainty was given to all the areas calculated by the automated image processing and a 2% uncertainty was associated with the perimeters. Consequently, the uncertainty of the hydraulic diameter for each channel was estimated to be 6%, based on a propagation of uncertainties analysis. The sequential perturbation method described by Figliola and Beasley [58] was used to calculate the uncertainty of the mean area and perimeter. Because a large number of channels

(1,058) were measured, the uncertainties in the mean area, mean perimeter, and mean hydraulic diameter were very small (on the order of 0.1–0.2%).

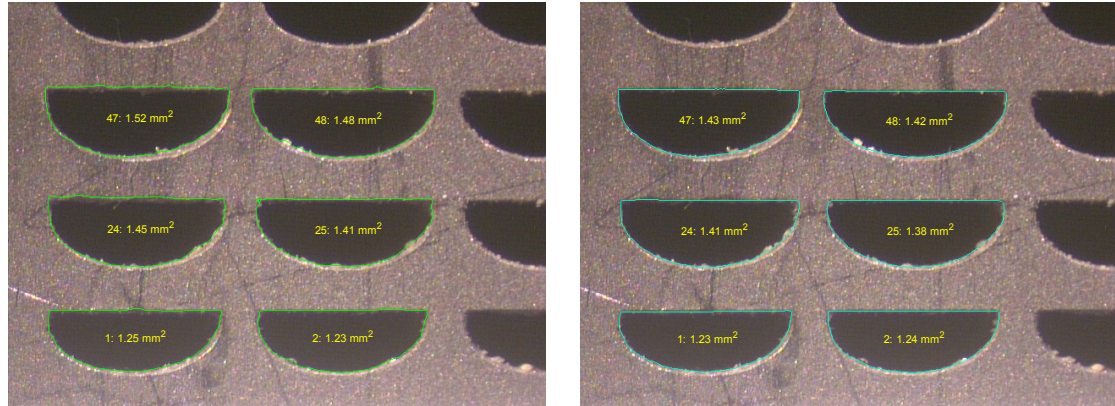


Figure 6.47. A comparison between the results of the automated outlining (left) and the manual outlining (right) methods

Table 6.1. Area and perimeter results of the two outlining methods

Channel Number	Area			Perimeter		
	Automated Outline	Manual Outline	% Difference	Automated Outline	Manual Outline	% Difference
1	1.25	1.23	1.6%	4.91	4.96	1.1%
2	1.23	1.24	0.6%	4.84	4.91	1.5%
24	1.45	1.41	3.4%	5.15	5.13	0.3%
25	1.41	1.38	1.7%	5.10	5.13	0.4%
47	1.52	1.43	5.9%	5.31	5.24	1.3%
48	1.48	1.42	4.1%	5.27	5.24	0.5%

A small improvement that could be implemented in the automated program is to remove excess pixels that are above the fitted line at the top of the channel. This would reduce the amount of influence of streaks or scratches that are present above the channels. As can be seen in the left image of Figure 6.47, the current scheme may produce channel outlines that are slightly above the opening of the channel. After reviewing all of the processed images, it

appears that this behavior affected many of the channels, but mostly to a lesser degree than the ones shown here.

6.2.2 Uncertainties in Profilometer Processing

The limitations and possible sources of error in the profilometer processing can be separated into two categories: those that arise from the measurement processes and ones that come from the calculations. A possible source of error from the measurement process is in the step where the 3-D profile data was collapsed into 2-D data. If the length of the channels is not parallel to the axis that is collapsed in the averaging, then the channel profile will be altered.

This effect is illustrated in Figure 6.48.

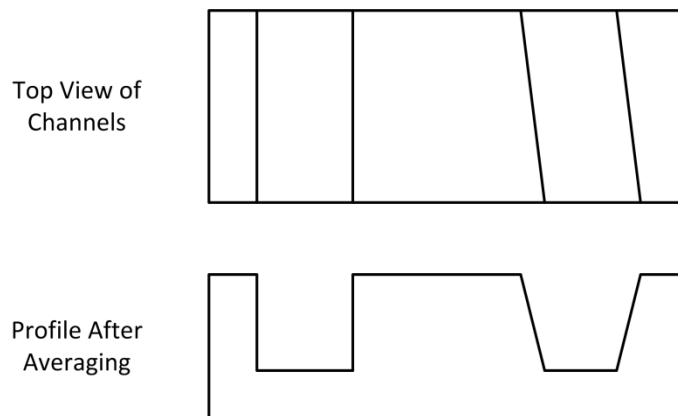


Figure 6.48. A comparison between the average profiles of a straight channel and an angled channel

Visual aids on the user interface of the ZeScope program can help position the shims quickly and easily on the measurement stage to minimize this effect. Another potential source of error is the amount of noise present in the scan. Noise can affect the detection of important features used in the calculation process, such as the shim tops and their edge points. Fortunately, much of the noise is taken out of the data by the post processing steps after data acquisition.

Scanning a larger area along the length of the channel, therefore creating more data to average, may help to reduce noise in the 2-D profile.

Potential sources of error and limitations to the profile processing may also come from the calculations. One limitation in the process comes from the use of line fits. Referring to Figure 6.49, fitting straight lines on the shim tops leaves a small area that is excluded in the calculations.

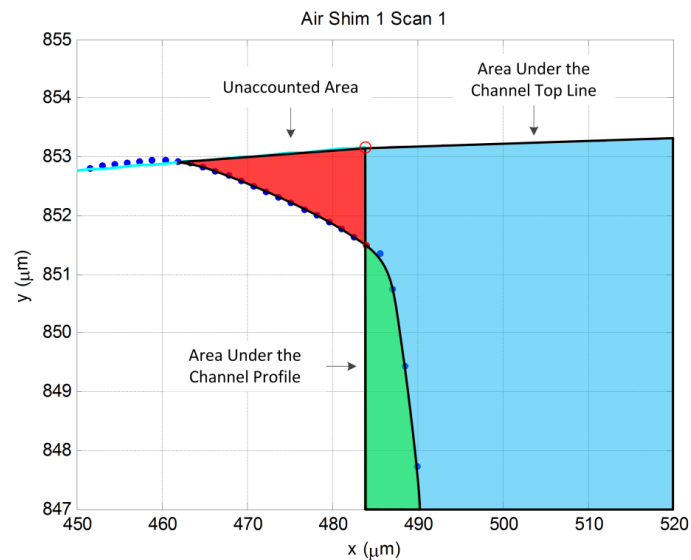


Figure 6.49. The unaccounted area in a shim profile

This effect happens when the fitted line is above the profile point that has been identified as the end of the shim top. The unaccounted area increases as the roundedness at the shim top becomes more pronounced. This behavior is a combination of two factors: the physically rounded shim tops produced by the chemical etching process, and the detection and definition of the shim tops in the computations. Another situation that happens less frequently is when the line fit is lower than the shim top edge. Both situations affect the calculations of area,

perimeter, and hydraulic diameter. More accurate measurements could be obtained by improving the identification of the shim top edge point on the profile. However, it is unlikely that the end result will significantly change, given the scale of the profiles.

Since the profile measurements taken on the ZeScope were highly accurate (0.01 nm repeatability), the primary contribution to the uncertainty was likely from the calculations. It is important to note that these measurements were taken on unbonded shims. The profiles may be different after the high temperature and high pressure bonding process. Therefore, a quantitative uncertainty estimate was deemed less important for this analysis than for the Microscope Image Analysis.

6.2.3 Uncertainties in Thermal Performance

The uncertainty of each measured variable was calculated using

$$u = \sqrt{u_{bias}^2 + (t \times u_{pres})^2} \quad (6.14)$$

where u_{bias} is the bias error, t is the Student's-t distribution factor, and u_{pres} is the precision error. The bias error of a measured variable consisted of the inherent limitations of the measuring instrument and the curve fit error of the calibration curve, if applicable. Calibration curves were all of the form

$$y_c = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \quad (6.15)$$

The curve fit error of a calibration curve, S_{yx} , is

$$S_{yx} = \sqrt{\frac{\sum_{i=1}^N (y_i - y_{ci})^2}{\nu}} \quad (6.16)$$

where ν is the degrees of freedom based on the number of data points and the order of the polynomial curve fit, $\nu = N - (n - 1)$. The quantity $y_i - y_{ci}$ is the difference between the measured variable and the curve fit prediction. The curve fit error is essentially a modified standard deviation. The Student's-t statistic for a 95% confidence interval was taken as 1.96 for all measurements because the number of measurement samples was large (> 60), hence a large degree of freedom. The precision error was taken as the standard deviation of the measurements, usually multiplied by the slope of the calibration curve if necessary. The mathematical representation of the precision error is shown below

$$u_{pres} = \left. \frac{dy}{dx} \right|_{curve} \times S_x \quad (6.17)$$

Table 6.2 lists the average relative uncertainty for the measured variables.

Propagation of the uncertainties to a derived quantity, such as the fluid energies and the effectiveness, was calculated using the root-sum-square method [58]

$$u_y = \pm \sqrt{\sum_i \left(\left. \frac{\partial y}{\partial x_i} \right|_{x=\bar{x}} \times u_{x_i} \right)^2} \quad (6.18)$$

The average relative uncertainties of the calculated quantities are summarized in Table 6.3.

Specific uncertainties of measured and calculated quantities are presented as error bars in the results sections.

Table 6.2. Average relative uncertainty of measured quantities

Measured Variables		
Measurement		Average Relative Uncertainty
Flow Rates	Air Volumetric Flow Rate (m ³ /s)	2%
	Oil Volumetric Flow Rate (m ³ /s)	2%
Pressures	Air Manifold Pressure (kPa)	4%
	Oil Pump Pressure In (kPa)	1%
	Oil Pump Pressure Out (kPa)	1%
	HRU Oil Pressure In (kPa)	1%
	HRU Oil Pressure Out (kPa)	1%
Temperatures	Manifold Temperatures (°C)	1%
	HRU Air Outlet Temperature (°C)	3%
	Air After Heaters Temperature (°C)	1%
	Ambient Air Inlet Temperature (°C)	9%
	HRU Oil Inlet Temperature (°C)	4%
	HRU Oil Outlet Temperature (°C)	1%
	After Flat Plate HEX Temperature (°C)	4%

A measurement that had a high relative uncertainty was the ambient air inlet temperature. This measurement was located at the inlet of the air blower and was used to calculate the air density near the flow meter. This seemingly high uncertainty in the measured temperature did not significantly affect the calculations because the density of air is not very sensitive to small changes in temperature. The average relative uncertainty in the velocity measurements is omitted from this table as it is presented in the following section. The primary contributor to the uncertainty of the oil energy was the oil mass flow rate.

Table 6.3. Average relative uncertainty of calculated quantities

Calculated Quantities		
Calculated Quantity		Average Relative Uncertainty
Fluid Properties	Average Air Temperature [Manifold & Outlet] (°C)	1%
	Average Air Specific Heat (kJ/kg-K)	0.04%
	Air Inlet Density (kg/m ³)	1%
	Average Oil Temperature [HRU Inlet & Outlet] (°C)	1%
	Average Oil Specific Heat (kJ/kg-K)	1%
	Average Oil Density (kg/m ³)	0.2%
Mass Flows	Air Mass Flow Rate (g/s)	2%
	Oil Mass Flow Rate (g/s)	2%
Thermal Performance	Average Manifold Temperature (°C)	0.2%
	Air Energy (kW)	2%
	Oil Energy (kW)	5%
	Effectiveness (%)	1%
	Air Pressure Drop (kPa)	4%
	Oil Pressure Drop (kPa)	2%

6.2.4 Uncertainties in Velocity and Temperature Profiles

The positional accuracy and repeatability of the traverse stages was 23 μm and 5 μm , respectively, over the full travel range. Each stage had a linear encoder that was used in conjunction with the motor driver to position the stage precisely. The encoders had a resolution of 5 μm . Using the root-sum-square method from Figliola and Beasley [58], the uncertainty of travel was calculated as 25 μm .

On average, the uncertainty of the velocity measurements was 2 m/s. The velocity uncertainty was dominated by the level of fluctuations in the readings (standard deviation).

Where there was more flow and turbulence, the fluctuations were larger. The relative uncertainties varied widely. The uncertainty profiles of the velocity looked like the measurement profiles except with the magnitudes scaled down.

7 Results and Discussion

7.1 Microscope Image Analysis

Overall, the automated outlining method developed in the Microscope Image Analysis accurately traced the microchannel openings. An example of a very clean result is shown in Figure 7.1.

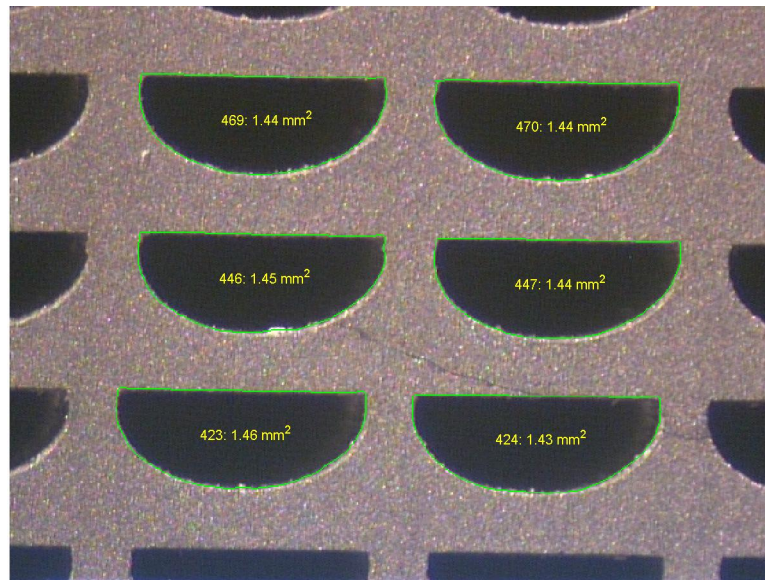


Figure 7.1. An example of one of the best results from the Microscope Image Analysis

The best outlining tended to be in the center of the face, where there were less markings and debris. One of the images with the most inaccurate outlines is the one that has been previewed in the analysis steps. It is shown again in Figure 7.2. Looking at channels 24 and 42 in Figure 7.2, it is apparent that the dark markings affected the outlining at the tops of the channels. This sort of behavior in the outline detection could be mitigated by excluding pixels that lie above the channel top line (the line mask). This could be implemented in the future as an

improvement to the analysis. This image illustrates that care should be taken to keep the sample surface free from any significant scratches or markings and debris.

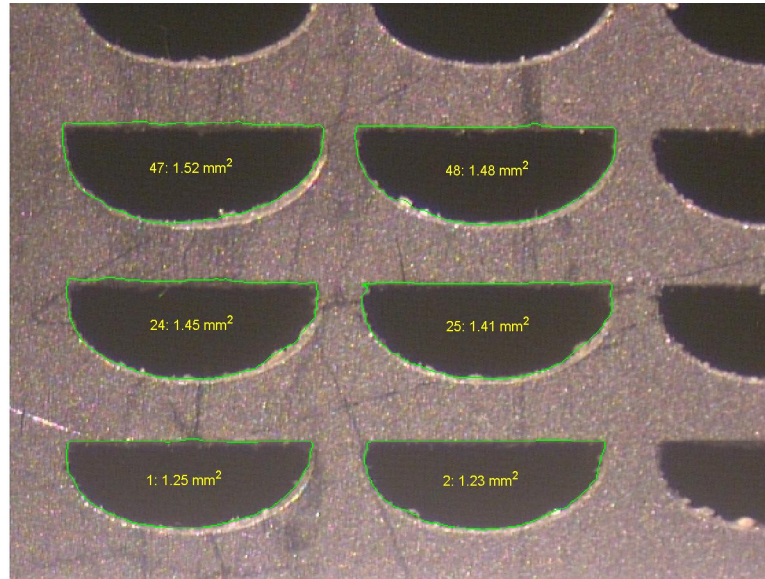


Figure 7.2. An example of one of the worst outlining results from the Microscope Image Analysis

As can be seen from some of the images (e.g. Figure 7.1) and also by visual inspection (e.g. Figure 3.9 in Section 3.4), the channel misalignment is very apparent. This misalignment can also be viewed in the velocity scans results, particularly in Figure 7.22 and Figure 7.23 in Section 7.3.2.1.

A plot of the air channel areas from the Microscope Image Analysis is shown in Figure 7.3. Plots of perimeter and hydraulic diameter are presented in Appendix A.1. Table 7.1 summarizes the results of the Microscope Image Analysis.

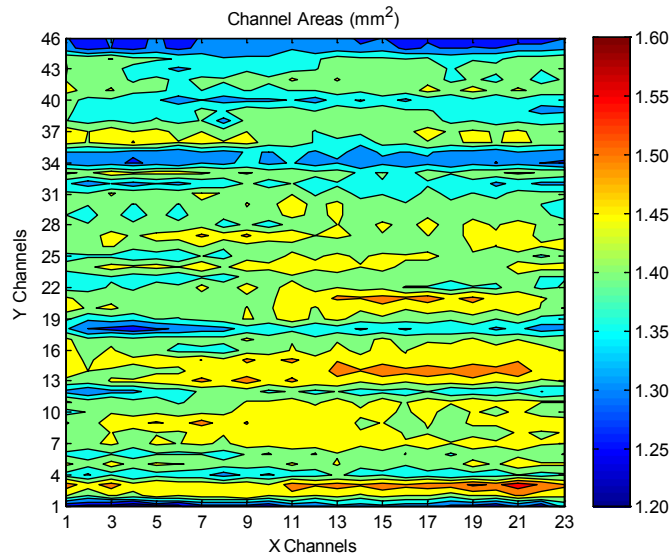


Figure 7.3. Air channel areas from the Microscope Image Analysis

Table 7.1. Statistical summary of the Microscope Image Processing results

	Design	Mean		Largest		Smallest		Standard Deviation
Quantity	Value	Value	% of Design	Value	% of Design	Value	% of Design	Value
Area (mm ²)	1.60	1.42	89%	1.58	99%	1.21	76%	0.056
Perimeter (mm)	5.60	5.06	90%	5.52	99%	4.80	86%	0.113
Hydraulic Diameter (mm)	1.14	1.12	98%	1.22	106%	1.00	88%	0.028

The cross-sectional channel areas were, on average, about 11% less than the design area. From Figure 7.3, the areas tended to be somewhat uniform along a shim, as indicated by the consistent coloring that spans horizontally. Vertically, the channel areas differed greatly. This indicates that while channel areas were uniform along a single shim, they varied significantly on different shims. This comes as no surprise when considering that the photochemical etching process was done in batches. Manufacturing parameters, such as the etching time, may have slightly changed from one batch to the next, creating deeper or shallower channels on a particular batch of shims. Variations in the shim thickness could also produce this effect.

Although the mean area and mean perimeter both deviated from the ideal dimensions by approximately 11%, the hydraulic diameter was very close to its design. This result will be revisited in the discussion of the thermal test data (Section 7.3.1). A comparison between the results of the Microscope Image Analysis and the results of the Profilometer Analysis is presented in the next section.

7.2 Shim Profile Analysis

Profiles of the air and oil microchannels are shown Figure 7.4 and Figure 7.5. As with the microscope images, the air shim channel profiles show a bottom surface that is fully rounded.

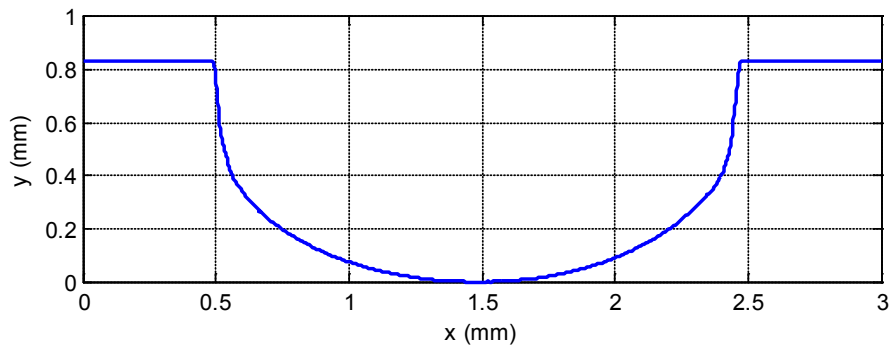


Figure 7.4. Air channel profile

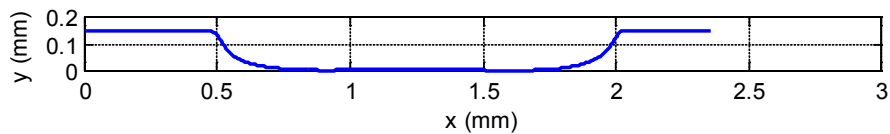


Figure 7.5. Oil channel profile

The air and oil channel areas are plotted in Figure 7.6 and Figure 7.7, and the results are summarized in Table 7.2. The average air channel area and perimeter were smaller than the

design by 15% and 12%, respectively. The mean hydraulic diameter, however, was not far from the design, at only 4% less than intended.

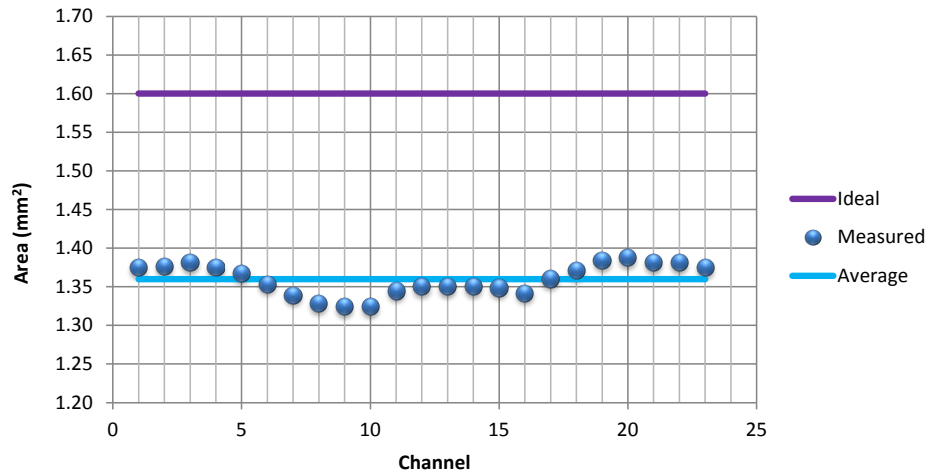


Figure 7.6. Air shim cross-sectional channel area results

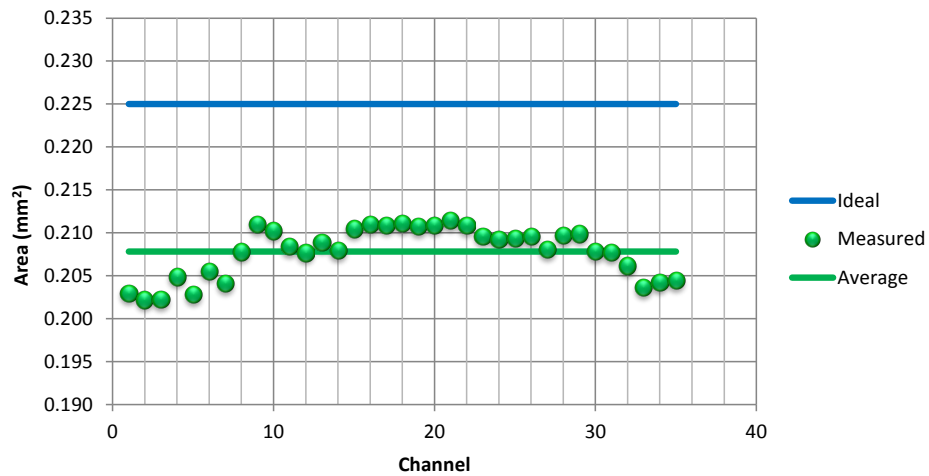


Figure 7.7. Oil shim cross-sectional channel area results

The oil channels were closer to their design than the air channels. In particular, the mean perimeter of the oil channels was very close to the design. This is likely due to the smaller

fillets at the corners. Additional plots of perimeter and hydraulic diameter can be found in Appendix A.2.

Table 7.2. Summary of Profilometer Processing results

	Air Shim Channels			Oil Shim Channels		
	Design	Mean	% of Ideal	Design	Mean	% of Ideal
Area (mm ²)	1.60	1.36	85%	0.225	0.208	92%
Perimeter (mm)	5.60	4.94	88%	3.30	3.23	98%
Hydraulic Diameter (mm)	1.14	1.10	96%	0.273	0.258	94%

The results from the Microscope Image Processing and the Shim Profile Analysis agree very well for the measurements of the air channels. This agreement supports the validity of the two techniques, as they are both independent. The results of the Microscope Image Processing should be given precedence over the Shim Profile Analysis in terms of representing the air channels for two reasons. First, the measurements were taken from the fully assembled device, whereas the profilometer analysis used unbonded shims to determine the channel geometries. During the bonding process, the shims of the final device were exposed to high temperatures and pressures, which could have caused some deformation in the final channels. The present analyses do not give insight as to how the bonding process alters the final channel shape, but the methods and architecture developed in this study could be used to investigate this hypothesis in future work. Second, the sample size in the Microscope Image Analysis is much greater than the sample size of the Shim Profile Analysis (1,058 channels compared to 35). Thus, the results of the Microscope Image Analysis should be considered more representative of the air channel geometries than the results of the Shim Profile Analysis.

7.3 Heat Exchanger Performance Tests

7.3.1 Thermal Tests

For all thermal tests, the manifold thermocouples indicated a uniform air temperature distribution into the Heat Recovery Unit. Graphical results of the duty, effectiveness, air pressure drop, and oil pressure drop are presented in the following subsections. Additionally, comparisons of performance between cases with a warm oil inlet temperature (73 °C average) and a cold oil inlet temperature (53 °C average) are shown and discussed.

7.3.1.1 Duty

The duty is presented as the oil energy. Figure 7.8 through Figure 7.10 each display the duty for a constant oil flow rate. These tests were conducted with an oil inlet temperature near 73 °C; the closest to the design.

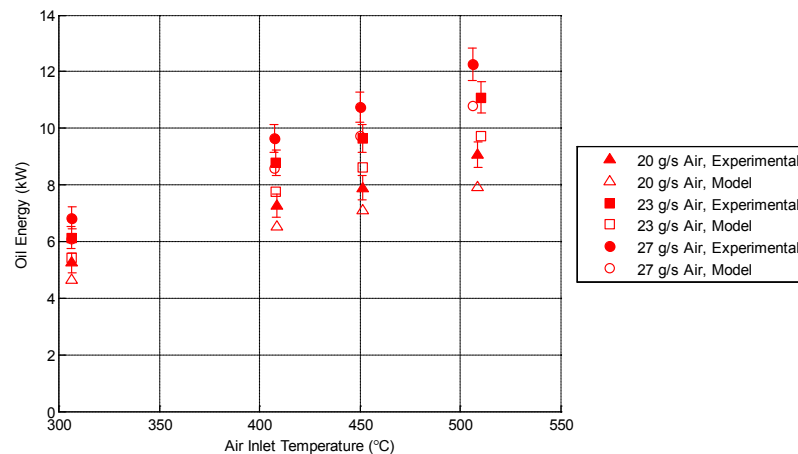


Figure 7.8. Duty for 40 g/s warm oil flow

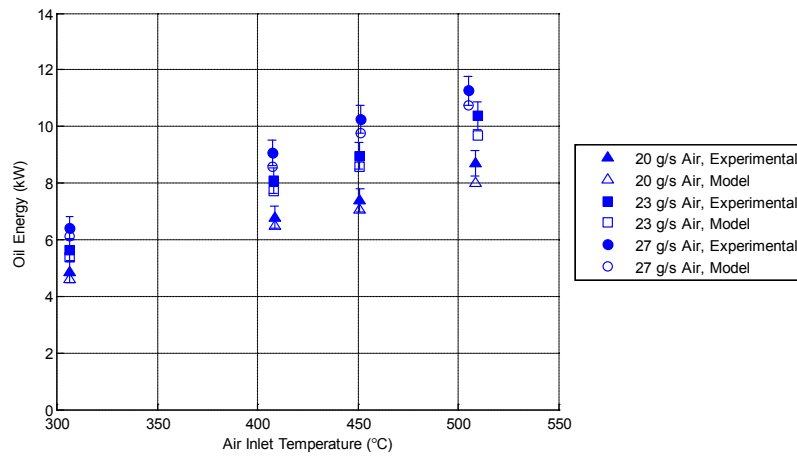


Figure 7.9. Duty for 47 g/s warm oil flow

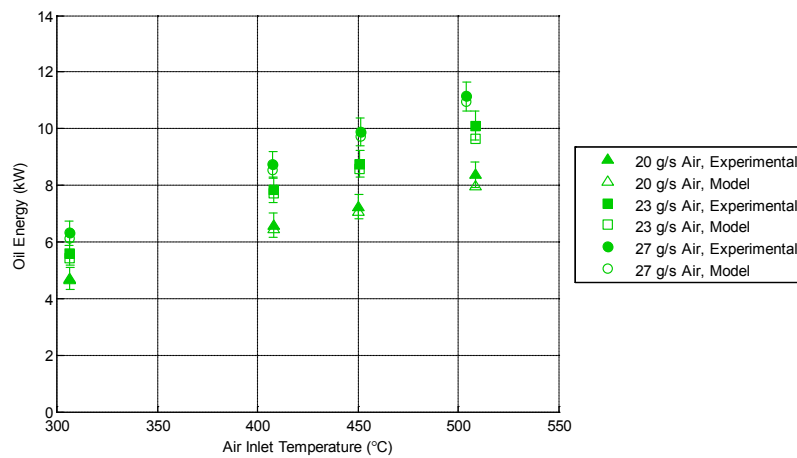


Figure 7.10. Duty for 55 g/s warm oil flow

For a constant oil mass flow rate, increasing the air inlet temperature increases the amount of energy picked up by the oil. Similarly, as the air mass flow rate increases, so does the oil energy. Increasing the air inlet temperature and the air mass flow rate effectively increases the available energy for exchange. As expected, if more energy is provided to the heat exchanger, then more energy is moved to the cooler fluid.

For 40 g/s oil mass flow rate (Figure 7.8), the experimental results are consistently higher than the model predictions, even when considering the uncertainty. However, for the cases of 47 g/s and 55 g/s of oil flow, the experimental heat transfers match the predicted heat transfer to within the estimated uncertainty. To determine if this behavior was caused by the experimental system or by the model, the oil energy and air energy were compared. In Figure 7.11, the oil and air energies are plotted in ascending order of the air energy.

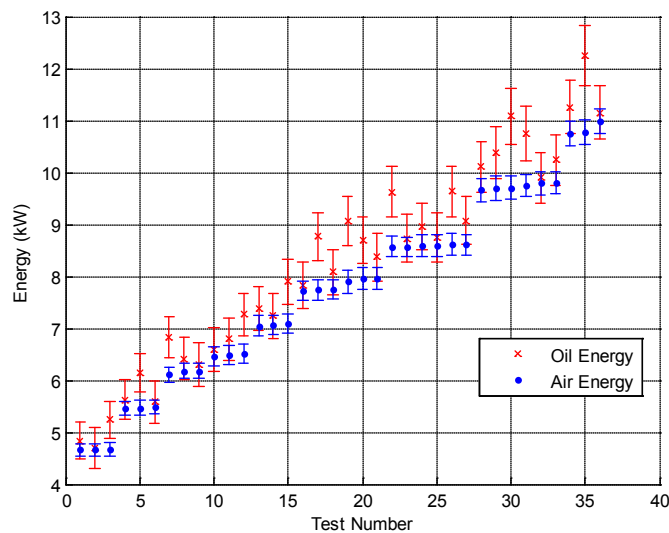


Figure 7.11. A comparison between the air energy and oil energy (warm oil cases)

The first notable attribute of this graph is that the oil energy is consistently higher than the air energy, neglecting the uncertainty. This is opposite of what is expected. In ideal heat transfer, the air and oil energies would match perfectly. In the real world, there are losses in heat transfer that make it impossible for all of the energy from the hot stream to be transferred to the cold stream. Therefore, the oil energy is expected to always be less than the air energy. When considering the uncertainties, two-thirds of the tests showed good agreement of the

energy balance. The remaining one-third, however, had oil energies above the air energies. An investigation revealed that for all of the cases with an inconsistent energy balance, the nominal oil flow rate was 40 g/s. This indicates that something during the testing could have affected the results, or that the oil catch and weigh calibration of the flow meter may be inaccurate at the lower flow rate. Consequently, the trend of significantly higher performance of the experimental data may not be accurate for the 40 g/s oil flow rate. The heat transfers for the other two oil flow rates matched well with the model, and the energy balances also agreed. Therefore, only the results of the 40 g/s oil seem to be affected.

With these considerations in mind, the effect of altering the oil mass flow rate is examined. Increasing the mass flow rate of oil seems to have little effect on the amount of heat transferred, for the range tested. Most likely, the differences in the tested oil mass flow rates were not large enough to see a significant change in heat transfer.

Overall, the model predicted the energy transfer very well. For the air energy, the model agreed with the experimental values almost exactly (less than 1% difference), which is why plots of the air energy are not shown here. The Microscope Image Analysis and the Shim Profile Analysis results indicated that the perimeters were smaller than the design. The perimeter determines the surface area for heat transfer. The heat transfer results indicate that the total surface area was not significantly different from the design, even though the channel perimeters were smaller. The hydraulic diameters of the air and oil channels were also measured to be close to the design. In the model, the hydraulic diameter is used to calculate the convection coefficient and is, therefore, directly tied to the heat transfer. Since the

hydraulic diameter and the heat transfer surface area were close to the design, the measured heat transfer matched the model prediction fairly well.

7.3.1.2 Effectiveness

Three figures of the effectiveness are presented for the three oil flow rates. These plots also correspond to the tests with the warm oil inlet temperature.

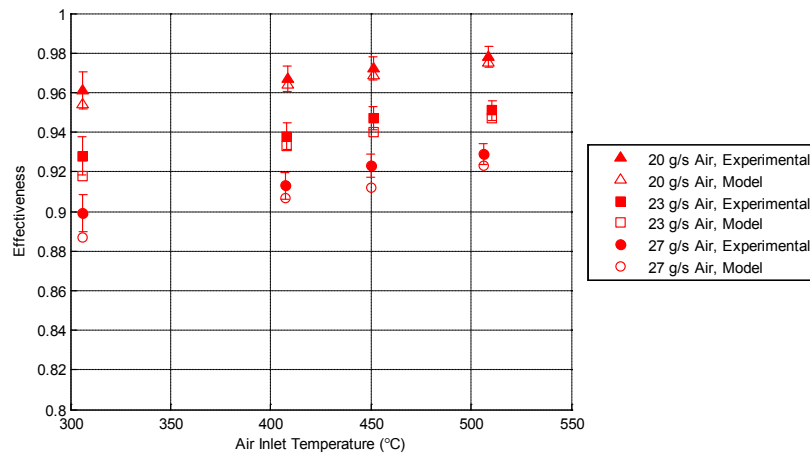


Figure 7.12. Effectiveness for 40 g/s warm oil flow

The effectiveness increased slightly as the inlet air temperature increased. Unlike the duty, the effectiveness decreased when the air mass flow rate increased. Higher air mass flow rates reduce the amount of time the air spends in the heat exchanger, and thus there is less time for energy to transfer. This results in lower temperature differences and effectiveness.

The effectiveness seems to be insensitive to oil flow rate for these conditions. Nearly all of the experimental data matches the model predictions within the uncertainty. For those that do not fall within the uncertainty, the difference is minimal. The very slight discrepancy in the effectiveness for a few cases may be due to the air outlet temperature measurement.

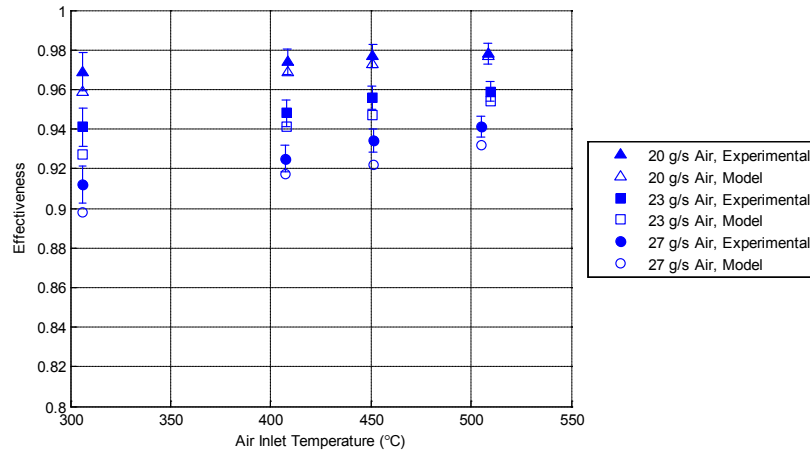


Figure 7.13. Effectiveness for 47 g/s warm oil flow

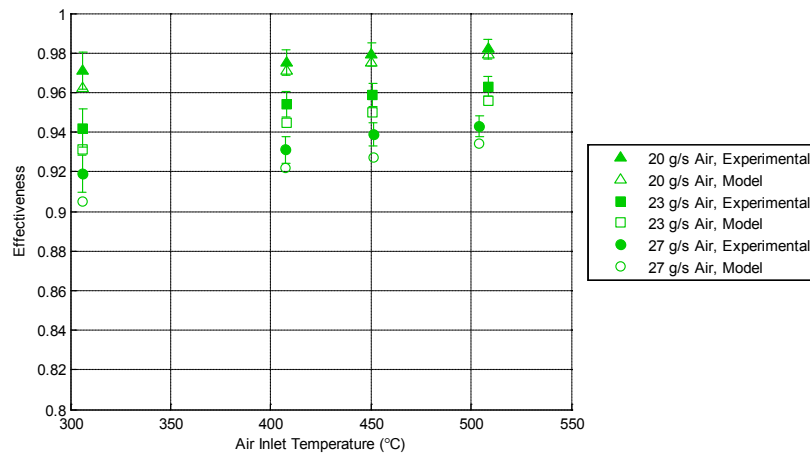


Figure 7.14. Effectiveness for 55 g/s warm oil flow

The air outlet temperature was only measured at a single location for the thermal tests. The thermocouple was placed at the center of the air exhaust plane, 1 mm away from the face of the HRU. Since it is unlikely that the thermal profile was not uniform (see Section 7.3.3), as the model presumes, the location of the air outlet temperature measurement may influence the resulting effectiveness. If the temperature at the center of the channel array represents a

relative maximum, then the calculations of effectiveness and heat transfer based on this temperature will underestimate the actual performance.

In general, the results of the effectiveness confirm that the Heat Recovery Unit performed as the model predicted. Similar to the heat transfer, the measured channel geometries did not greatly affect the effectiveness.

7.3.1.3 Air Pressure Drop

The air pressure drop is plotted versus the air mass flow rate in Figure 7.15. The plot shows that the air pressure drop was significantly more than what the model predicted for all the test cases. The general trend of increasing pressure drop as flow rate increases is displayed by both the experimental data and the model prediction. The pressure drop also increases for an increase in the air temperature, as expected, because the viscosity of air increases with the temperature.

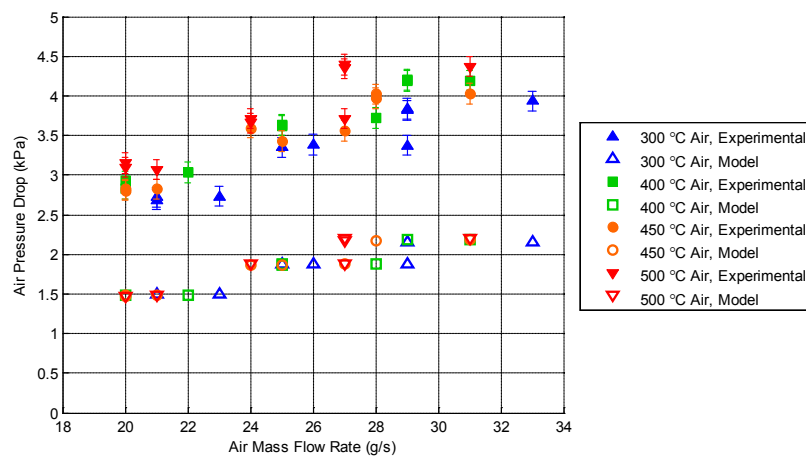


Figure 7.15. Air pressure drop

Several factors could be contributing to the discrepancy of the air pressure drop. First, the cross-sectional areas of the air channels were measured to be smaller than the design. This means the flow area was reduced, which would increase the pressure drop. Second, entry length effects may not be negligible. A calculation of the hydrodynamic entry length, based on the design flow rate, design temperature, and the measured channel geometry, shows that the entry region is approximately 17% of the total channel length. Table 7.3 lists the design conditions and the resulting calculation of the entry length.

Table 7.3. Reynolds number and entry length for the air channels at the design flow conditions and measured geometry

Air Channels	
Total Inlet Mass Flow Rate (g/s)	30
Inlet Temperature (°C)	488
Reynolds Number per Channel	574
Hydrodynamic Entry Length (mm)	36
Total Channel Length (mm)	210
Entry Length / Channel Length (%)	17%

Because the friction factor in the entry region is higher than in the fully developed flow regime, the pressure drop at the entrance is higher. This could be contributing to the difference in pressure drop between the measured data and model predictions. The model neglects entrance effects.

Another potential cause for the discrepancy could be that the friction factor correlation in the model may not be completely accurate for the shape of the microchannels in the HRU. The correlation was based on a rectangular channel, but the channels in the HRU are nearly half rectangular and half circular, as seen from the Microscope Image Analysis and the Shim Profile Analysis.

The use of laminar flow correlations was appropriate because the mean Reynolds number between the inlet and outlet was well below the transition to turbulence. The mean Reynolds number of the air stream ranged from approximately 500 to 1,000.

It is noteworthy to comment that the measured air mass flow rates seem to be almost evenly distributed across the range, instead of being clustered in the groups of 20 g/s, 23 g/s, and 27 g/s. This can be explained by how the air blower was controlled. Data was recorded for constant blower frequencies (i.e. 25 Hz, 30 Hz, and 34 Hz). This method did not necessarily ensure that the same air mass flow rates were achieved between the test cases. A better method would have been to calculate the mass flow rate in real-time with LabVIEW, instead of in the post-processing. Then the blower frequency could have been adjusted to reach the desired mass flow rate. The reported air mass flow rates (20 g/s, 23 g/s, and 27 g/s) are averages taken from constant blower frequencies.

7.3.1.4 Oil Pressure Drop

The oil pressure drop is plotted versus oil inlet temperature in Figure 7.16 because it is easier to view the trends than if it were plotted as pressure drop versus flow rate. The experimental data shows that the oil pressure drop tends to decrease as the temperature increases. This is not surprising, as the viscosity of Paratherm NF is fairly sensitive to changes in temperature in this region. The viscosity decreases as the oil gets warmer, thus decreasing its flow resistance. As the oil mass flow rate increases, the pressure drop also increases. This is also expected since increasing the mass flow rate is effectively forcing more fluid through a channel with a constant cross-sectional area. Doing so requires more power and a larger pressure differential.

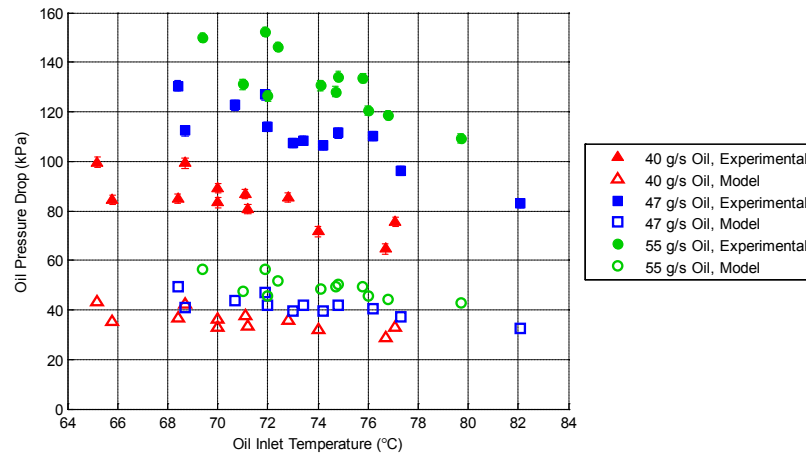


Figure 7.16. Oil pressure drop

The magnitude of the experimental data is significantly different from the model predictions by consistently more than a factor of two. Some of the same explanations discussed in the previous section for the inconsistency between the experimental and model results of the air pressure drop are also valid for the oil pressure drop. Namely, the smaller cross-sectional area and shape of the oil channels could be causing a conflict with the measured data and the model.

The hydrodynamic entry length of the oil channels, under the design conditions and the measured geometry, is much smaller than the total channel length at approximately 0.3% of the channel length. Table 7.4 shows the design conditions and the calculated entry length of the oil channels. Entrance effects, at least in the core region of the heat exchanger, are not likely contributing to the discrepancy of the oil pressure drop.

The most probable reason why the model and experimental oil pressure drops do not match is because the model does not account for any of the header or plenum features. There

are significant constrictions at the inlet and outlet oil ports that could be greatly increasing the pressure drop.

Table 7.4. Reynolds number and entry length for the oil channels at the design flow conditions and measured geometry

Oil Channels	
Total Inlet Mass Flow Rate (g/s)	46
Inlet Temperature (°C)	100
Reynolds Number per Channel	12
Hydrodynamic Entry Length (mm)	0.2
Total Channel Length (mm)	60
Entry Length / Channel Length (%)	0.3%

Also, the plenum area between the two passes is completely unaccounted for in the calculations. The oil is allowed to mix in this region, and it must pass through a constriction in the middle of the plenum. These features are very likely causing the disagreement between the measured pressure drop and the model pressure drop.

7.3.1.5 Oil Inlet Temperature Comparison

In this section, the effects of the warm oil inlet and cold oil inlet temperatures are compared. The next two figures are of the duty and effectiveness for 47 g/s oil flow rate and oil inlet temperatures of 73 °C (warm oil) and 53 °C (cold oil). Additional plots for the other two flow rates can be found in Appendix A.4.

Figure 7.17 shows that the cold oil cases have marginally higher heat transfers. It is expected that a colder oil inlet temperature will increase the amount of heat transfer because the temperature gradient between the hot air and the cold oil is larger. The inlet oil temperatures in this study were not far enough apart to show any significant difference in the heat transfer within the uncertainty.

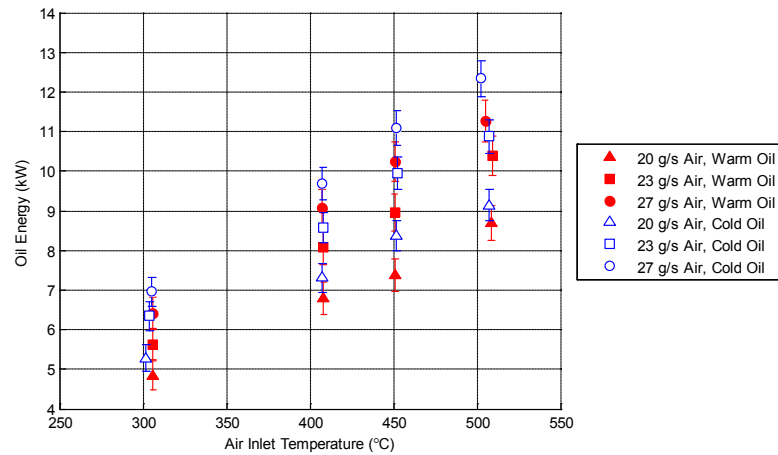


Figure 7.17. Duty for 47 g/s oil at inlet temperatures of 73 °C (warm) and 53 °C (cold)

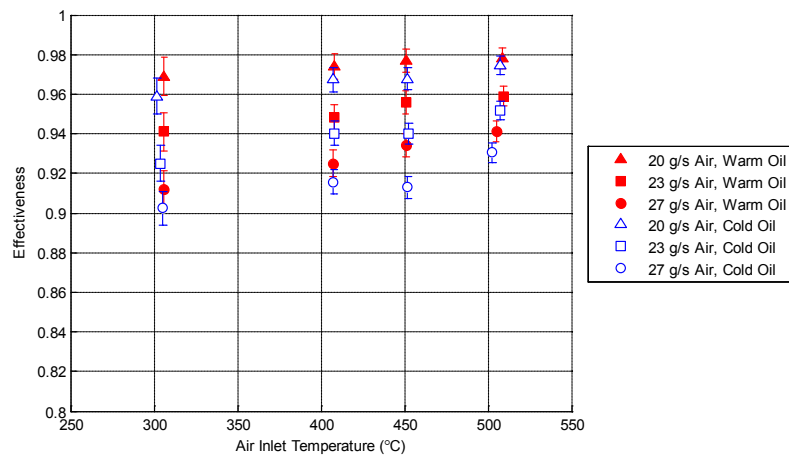


Figure 7.18. Effectiveness for 47 g/s oil at inlet temperatures of 73 °C and 53 °C

The effectiveness decreased slightly when the inlet oil temperature was lowered. When lowering the inlet oil temperature, the potential for heat transfer may increase, but the actual heat transfer may not increase to the same proportion. The effectiveness can be interpreted as a measure of how well the heat exchanger is able to bring the hot fluid down to the lowest

temperature of the cold fluid (the inlet temperature). By lowering the inlet temperature of the cold fluid, the heat exchanger must work harder to bring down the temperature of the hot fluid. Heat exchangers are limited by their finite temperature differences, finite surface areas, and also by the lowest heat capacity of the working fluids; not to mention insulation losses. These limitations prevent heat exchangers from transferring all of the heat from one fluid to the other and bringing the hot fluid down to the cold fluid inlet temperature. Therefore, lowering the oil inlet temperature caused a slight increase in heat transfer (though not significant within the range of the uncertainty), but decreased the effectiveness.

Upon examination of the oil pressure drop of the cold oil cases in comparison to the warm oil cases, it is apparent that the cold oil inlet temperature is not a good choice for operation because of the high pressure drop. Figure 7.19 illustrates the difference between the two scenarios.

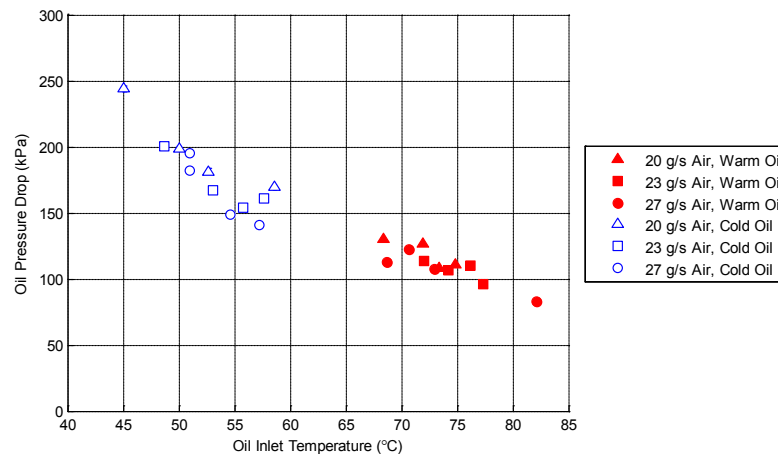


Figure 7.19. Oil pressure drop comparison for oil at 47 g/s

At low temperatures, the oil viscosity was very high, causing a larger pressure drop through the HRU. Warmer oil inlet temperatures are preferable for practical operation over the colder

oil temperatures because the effectiveness and pressure drop for the warmer temperatures are better.

The air pressure drop was not significantly affected by the change in oil inlet temperature.

7.3.2 Hot Wire Tests

All of the velocity measurements were taken with unheated air (25 °C inlet) at a mass flow rate of 27 g/s. The oil pump was off. For these conditions, and using the results from the Microscope Image Analysis for the average cross-sectional area, the theoretical velocity through a single air channel is approximately 16 m/s.

7.3.2.1 Horizontal and Vertical Scans

Several plots of the horizontal and vertical velocity scans are presented in this section. Figure 7.20 shows 3-D plots of each scan; Figure 7.21 shows side views of the scans; and contour plots of the horizontal and vertical scans are shown in Figure 7.22 and Figure 7.23. Wherever possible, the x and y directions were formatted to be equally scaled, but one-to-one scaling was not always accomplished. When interpreting the plots, it is important to note the axes scales. For the contour plots, the color scales are the same for comparison. It should also be noted that the origins for each scan are independent. Precise spatial locations are not directly comparable between scans.

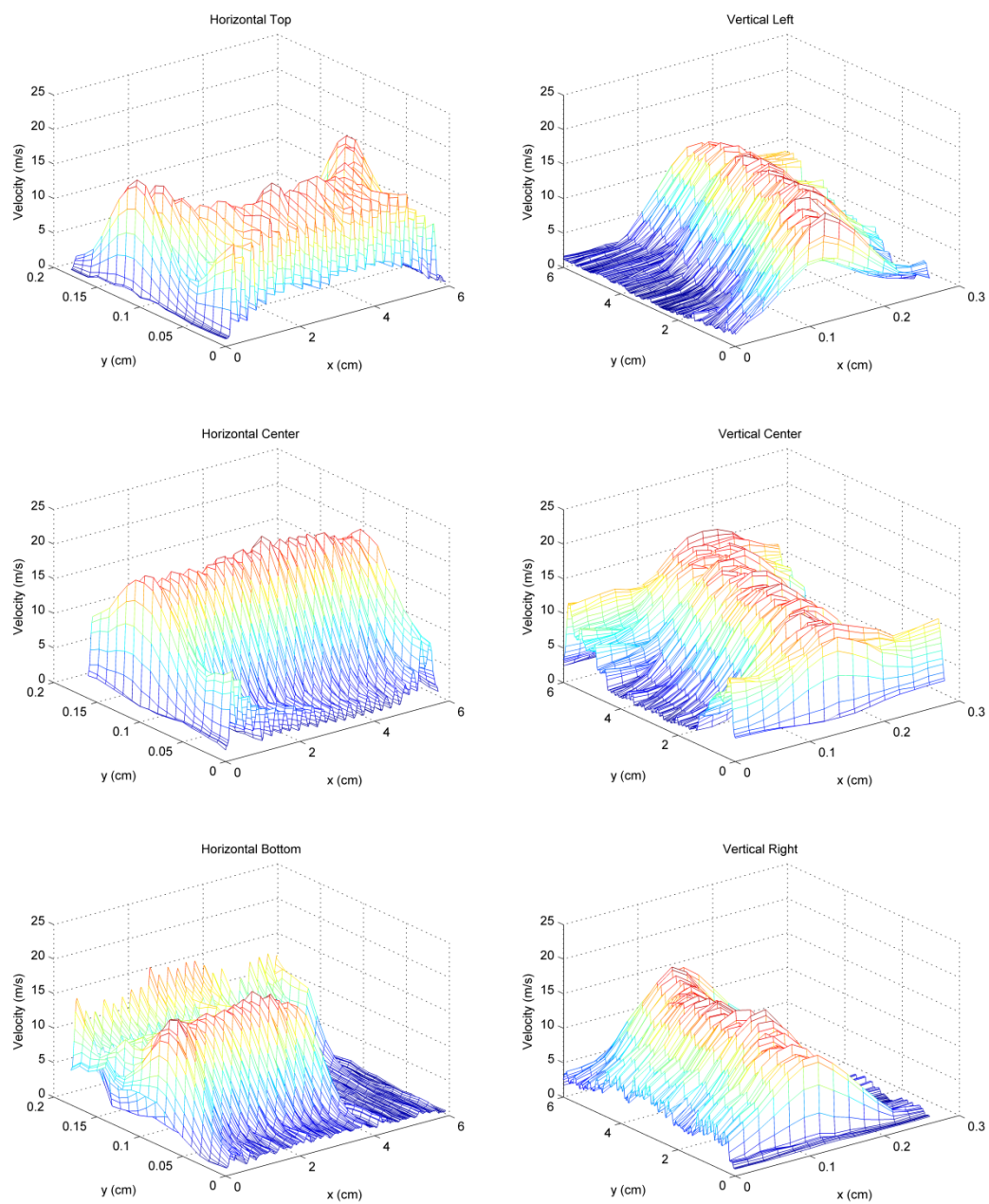


Figure 7.20. 3-D plots of horizontal and vertical velocity scans

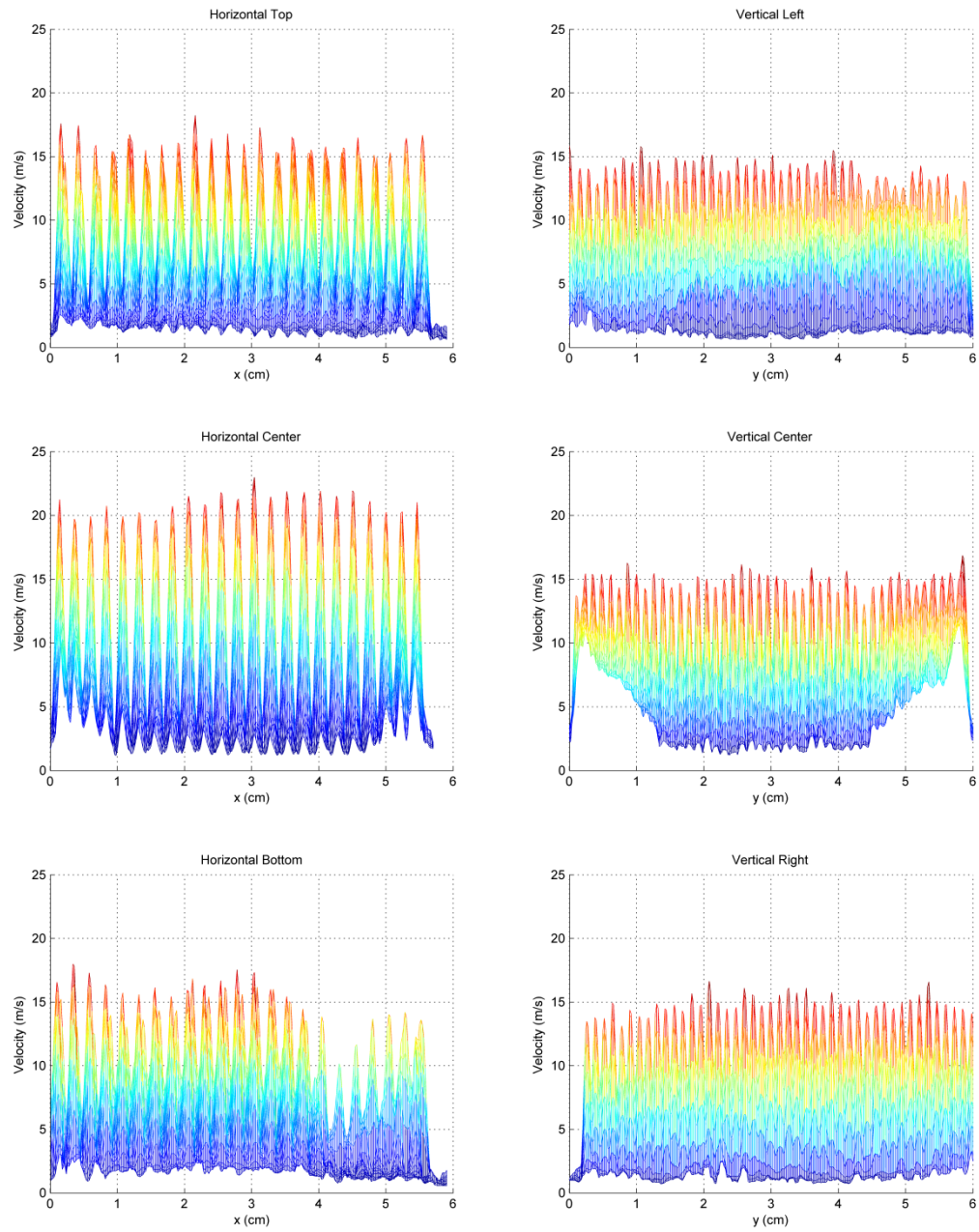


Figure 7.21. Side views of the horizontal and vertical velocity plots

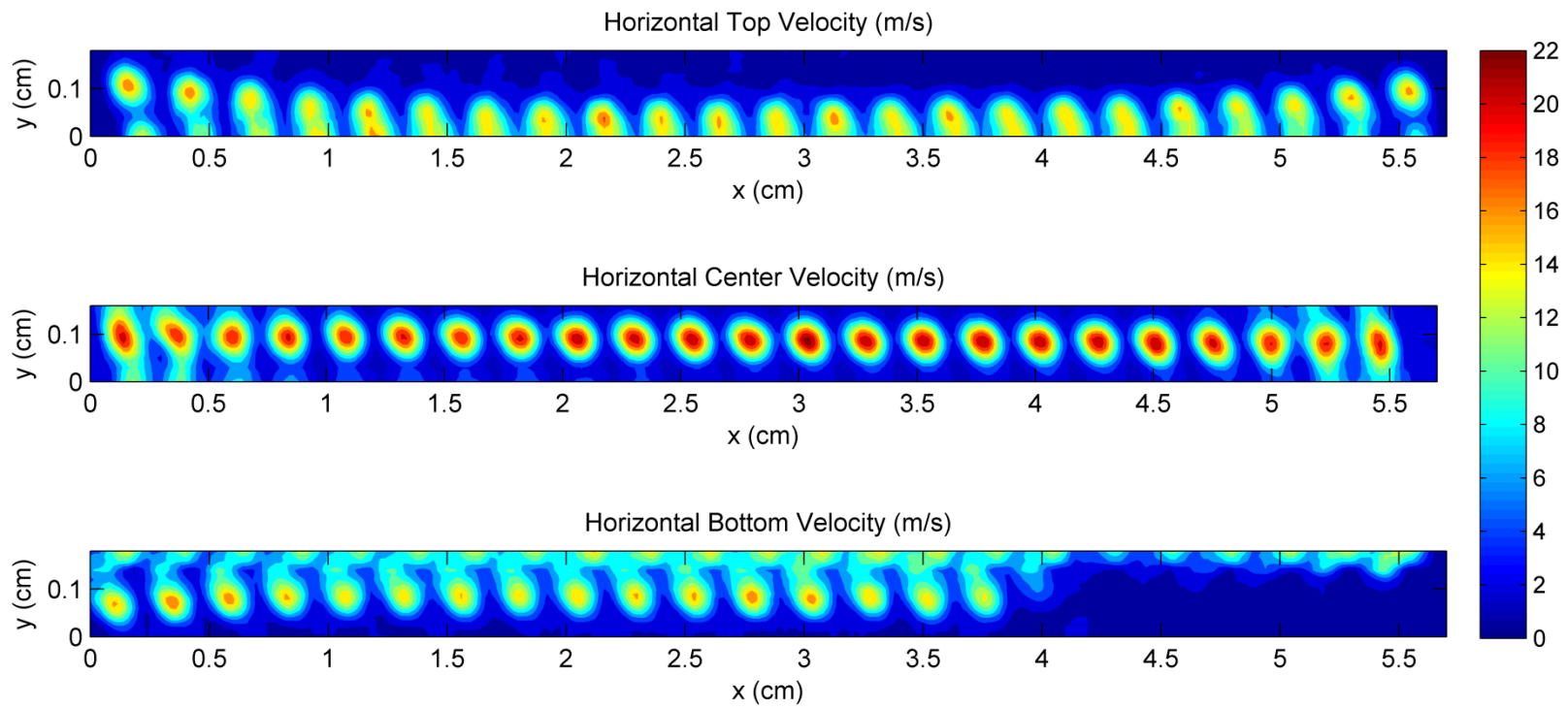


Figure 7.22. Contour plots of the horizontal velocity scans

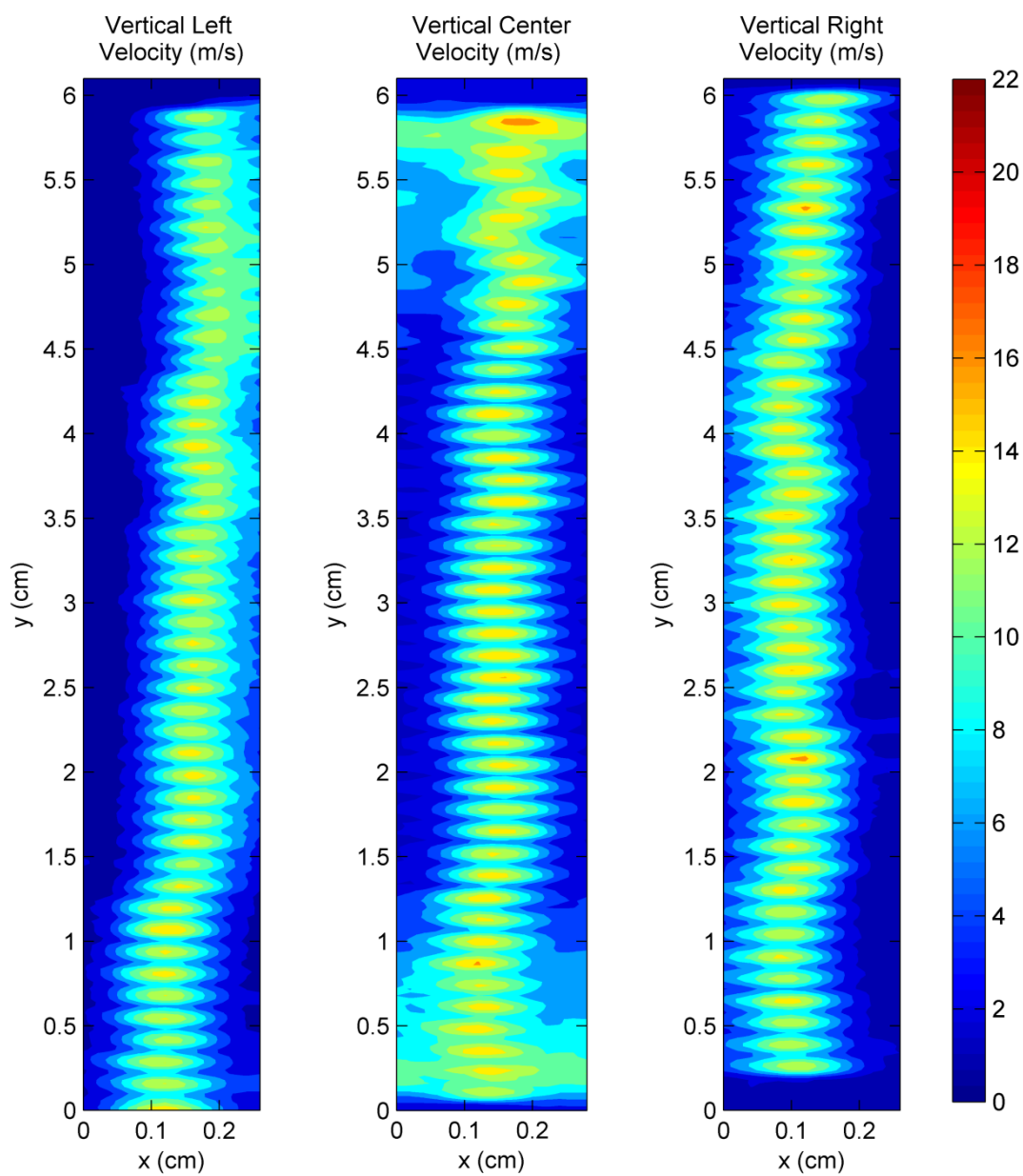


Figure 7.23. Contour plots of the vertical velocity scans

Velocity profiles of individual channels are distinguishable in the figures. Each figure shows that the velocity magnitudes were fairly uniform within the scan. For all scans except the horizontal center, the maximum velocity was near 15 m/s, which is close to what theoretical calculation predicted. The horizontal center scan, however, had measured velocities that peaked near 20 m/s. A peculiarity of the horizontal and vertical scans is that the peak velocities do not match where the scans should intersect. As discussed in Section 6.1.3, the reason why the scans do not match is likely due to the resolution of the traverses. The vertical scans were taken with a resolution of 200 μm in the x-direction and 200 μm in the y-direction, whereas the horizontal scans were taken with a resolution of 200 μm in the x-direction and 100 μm in the y-direction. The horizontal scans were deemed more accurate at characterizing the flow than the vertical scans because the resolution was finer. Even though the vertical scans showed uniform velocities, it is possible for the flow to be non-uniform because the velocity profiles may be not be fully characterized. The results of the window scans (see the following section) confirm that the resolution was an issue. The window scans were taken with the highest resolution of all the traversing schemes at 100 μm in both the x and y directions. The largest magnitudes collected for the window scans agree with the results of the horizontal scans.

From Figure 7.20 and Figure 7.22, it can be seen that there are some channels missing in the horizontal bottom scan. This could be due to some blockage in the channels. In Figure 7.23, the vertical right scan is missing a channel at the bottom, which confirms that there is no flow through the channel. Curiously, if the number of channels in the vertical center scan is counted, it is also one less than expected (45 of 46). Since the vertical scan appears to have

traversed beyond the flow region both above and below the channels, it is possible that some debris clogged one of the channels during that test. During the horizontal scan, the same channel may have been unclogged because there is no discontinuity in the center velocity readings. The vertical left, horizontal right, and horizontal center scans have all of their channels present.

The horizontal and velocity scans depict the poor channel alignment, which affected the velocity profile, but not the heat transfer.

7.3.2.2 Window Scans

Figure 7.24 and Figure 7.25 display the results of the center window scan and the bottom left window scan. The velocity profiles of the nine channels captured in the center window scan appear to be very uniform in both the velocity magnitudes and the distribution shape. Figure 7.25 shows that there was some mixing or turbulence in the regions outside of the channels. This is not surprising as the flow is not expected to be completely uniform at the corners or edges of the channel array.

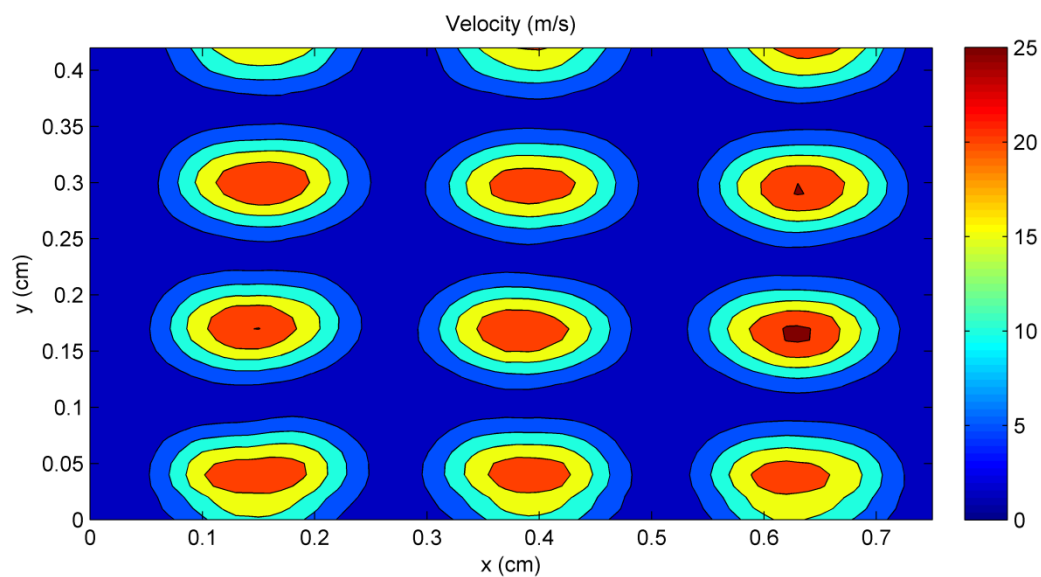


Figure 7.24. Velocity contour plot of the center window

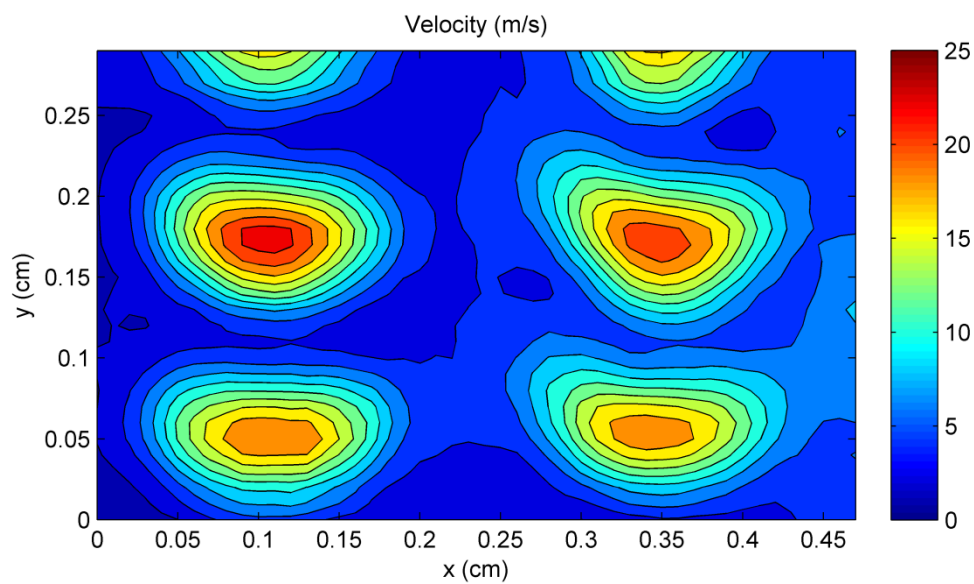


Figure 7.25. Velocity contour plot of the bottom left window

7.3.2.3 Full Field Maximum Velocity Interpolation

The velocity interpolation was meant to predict the full field flow behavior based on the maximum channel velocities measured from the traverses. Since the vertical center and vertical right scans only had 45 velocity peaks, maximum velocities were associated to only 45 channels in the vertical direction. Also, since the horizontal bottom scan showed some missing channels, the velocities used for the missing channels were from the measurements of the channels from the row above. The intermediate step of shifting the vertical scans to meet the horizontal scans can be seen in Appendix B. As stated in Section 6.1.3, the shifting of the vertical scans to meet the horizontal scans was deemed appropriate because the horizontal scan were better able to resolve the flow. The maximum velocity interpolation assumes that the channels are evenly spaced and does not consider misalignment.

The result of the maximum velocity interpolation is presented in Figure 7.26. The distribution shows high velocities in the center and lower velocities on the top and bottom. This is expected because the horizontal center scan had the highest maximum magnitudes of velocity across the entire scan. The horizontal uniformity of the velocity interpolation not only comes from the uniformity of the individual horizontal scans, but also from the shifted vertical scans. In the process of computing the full field interpolation, the maximum channel velocities of the vertical scans were flexible, and thus they formed to the trends of the horizontal scans. It is reasonable to expect, and see, that there was more flow in the center of the channel array. Edge effects in the manifold and in the piping upstream of the manifold may have affected the air flow in channels on along the periphery.

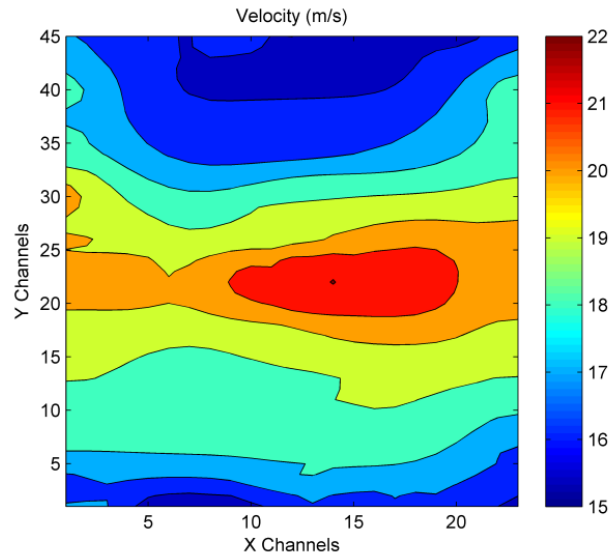


Figure 7.26. Full field maximum velocity interpolation result

7.3.3 Thermal Profile

A full field temperature profile was measured for one case of thermal loading. The testing conditions are tabulated in the table below. Measurements were taken at a resolution of 1 mm in both the x and y directions, approximately 2 mm away from the HRU face.

Table 7.5. Test conditions for the full field temperature profile

Controlled Parameters		Heat Exchanger Performance	
Air Mass Flow Rate (g/s)	27	Average Outlet Air Temperature (°C)	72
Inlet Air Temperature (°C)	451	Outlet Oil Temperature (°C)	155
Oil Mass Flow Rate (g/s)	56	Heat Transfer [Based on Oil Energy] (kW)	11.8
Inlet Oil Temperature (°C)	62	Effectiveness	0.97
		Air Pressure Drop (kPa)	3
		Oil Pressure Drop (kPa)	200

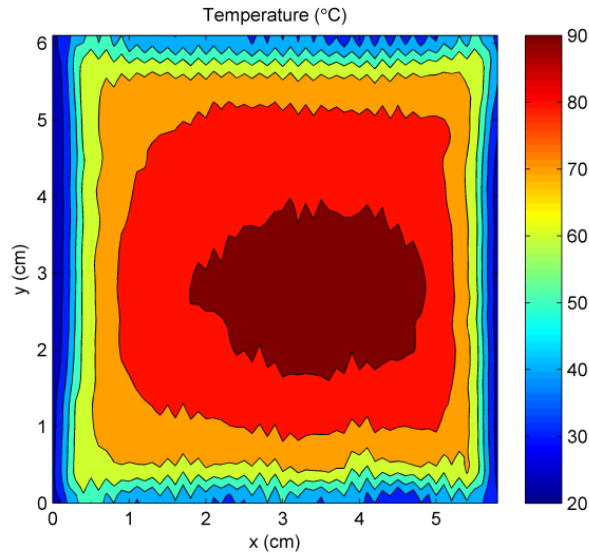


Figure 7.27. Full field temperature profile for conditions close to the design

The temperature profile's non-uniform shape is indicative of the heat transfer and flow of the Heat Recovery Unit. The air along the outer regions of the channel array was cooled more than the air in the center, resulting in lower temperatures on the sides and higher temperatures in the middle. The left and right sides of the profile are not mirrored. On the left side, the temperature contours show that there was slightly more cooling than on the right. This can be attributed to the geometry of the HRU or the flow distribution. Recall that, in this view, the header ports for the oil are located on the left side. The inlet oil starts cold on the left side then warms up as it travels to the right, picking up heat from the air stream. Consequently, the exhaust air was cooler on the left and warmer on the right because more heat transfer occurred on the left. It is also possible that the flow distribution affected the temperature profile. The velocity distribution for this thermally loaded condition was not measured. If the flow was

non-uniform, more heat transfer could occur in areas where the mass flow rate was lower, thus lowering the temperature.

8 Conclusion and Future Work

A cross-flow microchannel heat exchanger for applications in diesel engine waste heat recovery was geometrically characterized and thermally tested with hot air. Two analysis methods were developed to measure the microchannel geometry of the Heat Recovery Unit. The first method was the Microscope Image Analysis, which used image processing tools in MATLAB to determine the area, perimeter, and hydraulic diameter of the air channels. The second method, the Shim Profile Analysis, measured geometries from profilometer profiles of the air and oil shims. Both methods agree that the air channels had cross-sectional areas that were, on average, 11% less than the design. The average perimeter and hydraulic diameter of the air channels were 10% and 2% less than the design, respectively. The Shim Profile Analysis revealed that the oil channels were 8% smaller in area, 2% smaller in perimeter, and 6% smaller in the hydraulic diameter than their design.

The thermal test results showed that the heat transfer and effectiveness were not significantly affected by the difference between the actual channel sizes and the design. This was likely because the surface area was not significantly altered. The pressure drops of the air and oil streams were higher than the model prediction. The cross-sectional areas of the air and oil channels were smaller than their design, which could have caused the increase in observed pressure drop for both streams. Other factors that could have influenced the pressure drop, such as the model assumptions, were identified and discussed. The highest duty of the Heat Recovery Unit observed during the experimental tests was 12.3 kW and the highest effectiveness was 97.8%. The lowest recorded air pressure drop was 2.7 kPa and the lowest oil pressure drop was 64.6 kPa. Velocity measurements were collected for unheated air flowing

through the HRU. The resulting profiles, and an interpolation of the profiles, indicated that the flow was non-uniform. Channel misalignment was very apparent in the measured profiles. The misalignment did not seem to significantly affect the heat transfer performance. A temperature profile for a thermally loaded condition was measured and discussed.

The most significant improvement that could be made in the design of the next generation Heat Recovery Unit would be the reduction of the air and oil pressure drops. For the oil shim, opening the header and plenum regions could greatly reduce the pressure drop. Widening the channel areas of both the air and oil channels may also be beneficial in lowering the pressure drop, but the heat transfer performance would have to be optimized for the new geometry. Opening the air channels would help reduce the deposition of soot onto the channel walls when the HRU is operating with the diesel engine.

Both the Microscope Image Analysis and Shim Profile Analysis can be used for future measurements of microchannel geometries. The methodology and testing procedures developed for these analyses can be applied to almost any microchannel shape. The Microscope Image Analysis has the potential to be very useful for calculating the geometry of wavy corrugated and triangular microchannels. Slight alterations may need to be made, such as the shape outlining, but the general process would remain the same.

Several useful features could be added to the two geometric analyses. The Microscope Image Analysis could be extended to include estimations of channel misalignment based on the centroids of the channel openings. The potentially complex part about this feature would be relating the alignment of channels from one image to another. A relationship could be established by using the incomplete channels in the images. Both analyses could include

calculations of the radius of curvature of the corners. This would be useful for quantifying the characteristics of the etching process.

The thermal testing facility has the capability of characterizing the performance of future generations of the Heat Recovery Unit. The original intention of the traverse measurements was to compare exhaust velocity and temperature profiles for a thermally loaded condition. Unfortunately, time constraints and several broken hot wire probes prevented these experiments. However, the foundation for this work has been developed and can be utilized in the future.

Finally, the thermal performance data collected in this study could be used to develop or verify correlations between nondimensional numbers, such as the Nusselt number, Reynolds number, Number of Transfer Units (NTU), and effectiveness. The data can also be used to compare the HRU's performance on the diesel engine and the performance of the second generation Heat Recovery Unit.

References

- [1] “September 2011 Monthly Energy Review,” U.S. Energy Information Administration, Washington, D.C., DOE/EIA-0035(2011/09), Sep. 2011.
- [2] M. Moran and H. N. Shaprio, *Fundamentals of Engineering Thermodynamics*, 5th ed. Hoboken, NJ: Wiley, 2004.
- [3] B. F. Tchanche *et al.*, “Low-grade heat conversion into power using organic Rankine cycles – A review of various applications,” *Renewable and Sustainable Energy Reviews*, vol. 15, pp. 3963-3979, Oct. 2011.
- [4] T. C. Hung *et al.*, “A review of organic Rankine cycles (ORCs) for the recovery of low-grade waste heat,” *Energy*, vol. 22, pp. 661-667, Jul. 1997.
- [5] H. Wang *et al.*, “Experimental performance of a compliant scroll expander for an organic Rankine cycle,” *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, vol. 223, pp. 863-872, Nov. 2009.
- [6] K. J. Harada, “Development of a Small Scale Scroll Expander,” M.S. Thesis, Dept. Mech. Eng., Oregon State Univ., Corvallis, 2010.
- [7] H. Wang *et al.*, “Performance of a combined organic Rankine cycle and vapor compression cycle for heat activated cooling,” *Energy*, vol. 36, pp. 447-458, Jan. 2011.
- [8] V. Lemort *et al.*, “Testing and modeling a scroll expander integrated into an Organic Rankine Cycle,” *Applied Thermal Engineering*, vol. 29, pp. 3094-3102, Oct. 2009.
- [9] S. Quoilin *et al.*, “Experimental study and modeling of an Organic Rankine Cycle using scroll expander,” *Applied Energy*, vol. 87, pp. 1260-1268, Apr. 2010.
- [10] B. Saleh *et al.*, “Working fluids for low-temperature organic Rankine cycles,” *Energy*, vol. 32, pp. 1210-1221, Jul. 2007.
- [11] T. Yamamoto *et al.*, “Design and testing of the Organic Rankine Cycle,” *Energy*, vol. 26, pp. 239-251, Mar. 2001.
- [12] B. F. Tchanche *et al.*, “Fluid selection for a low-temperature solar organic Rankine cycle,” *Applied Thermal Engineering*, vol. 29, pp. 2468-2476, Aug. 2009.
- [13] E. H. Wang *et al.*, “Study of working fluid selection of organic Rankine cycle (ORC) for engine waste heat recovery,” *Energy*, vol. 36, pp. 3406-3418, May 2011.

- [14] "Waste Heat Recovery: Technology and Opportunities in U.S. Industry," U.S. Department of Energy, Industrial Technologies Program, Mar. 2008.
- [15] P. S. Patel and E. F. Doyle, "Compounding the Truck Diesel Engine with an Organic Rankine-Cycle System," SAE International, Warrendale, PA, 760343, Feb. 1976.
- [16] E. Doyle *et al.*, "Installation of a Diesel-Organic Rankine Compound Engine in a Class 8 Truck for a Single-Vehicle Test," SAE International, Warrendale, PA, 790646, Feb. 1979.
- [17] F. A. DiBella *et al.*, "Laboratory and On-Highway Testing of Diesel Organic Rankine Compound Long-Haul Vehicle Engine," SAE International, Warrendale, PA, 830122, Feb. 1983.
- [18] M. C. Brands *et al.*, "Vehicle Testing of Cummins Turbocompound Diesel Engine," SAE International, Warrendale, PA, 810073, Feb. 1981.
- [19] D. T. Hountalas *et al.*, "Improvement of bottoming cycle efficiency and heat rejection for HD truck applications by utilization of EGR and CAC heat," *Energy Conversion and Management*, vol. 53, pp. 19-32, Jan. 2012.
- [20] B. Agnew and M. Talbi, "Combined power and cooling, an analysis of the combined Diesel-absorption cycle," *Applied Thermal Engineering*, vol. 19, pp. 1097-1105, Oct. 1999.
- [21] M. Talbi and B. Agnew, "Energy recovery from diesel engine exhaust gases for performance enhancement and air conditioning," *Applied Thermal Engineering*, vol. 22, pp. 693-702, Apr. 2002.
- [22] H. Wang *et al.*, "Design study of configurations on system COP for a combined ORC (organic Rankine cycle) and VCC (vapor compression cycle)," *Energy*, vol. 36, pp. 4809-4820, Aug. 2011.
- [23] T. A. Davidson, "Design and Analysis of a 1 kW Rankine Power Cycle, Employing a Multi-vane Expander, for Use with a Low Temperature Solar Collector," Massachusetts Institute of Technology, 1977.
- [24] S. Probert *et al.*, "Design optimisation of a solar-energy harnessing system for stimulating an irrigation pump," *Applied Energy*, vol. 15, pp. 299-321, 1983.
- [25] X. D. Wang *et al.*, "Performance evaluation of a low-temperature solar Rankine cycle system utilizing R245fa," *Solar Energy*, vol. 84, pp. 353-364, Mar. 2010.

- [26] A. Bryszewska-Mazurek *et al.*, "Performance Analysis of a Solar-Powered Organic Rankine Cycle Engine," *Journal of the Air & Waste Management Association*, vol. 61, pp. 3-6, Jan. 2011.
- [27] S. Quoilin *et al.*, "Performance and design optimization of a low-cost solar organic Rankine cycle for remote power generation," *Solar Energy*, vol. 85, pp. 955-966, May 2011.
- [28] J. Bruno *et al.*, "Modeling and optimisation of solar organic Rankine cycle engines for reverse osmosis desalination," *Applied Thermal Engineering*, vol. 28, pp. 2212-2226, Dec. 2008.
- [29] A. M. Delgado-Torres and L. García-Rodríguez, "Analysis and optimization of the low-temperature solar organic Rankine cycle (ORC)," *Energy Conversion and Management*, vol. 51, pp. 2846-2856, Dec. 2010.
- [30] A. M. Delgado-Torres and L. García-Rodríguez, "Double cascade organic Rankine cycle for solar-driven reverse osmosis desalination," *Desalination*, vol. 216, pp. 306-313, Oct. 2007.
- [31] D. Manolakos *et al.*, "Design of an autonomous low-temperature solar Rankine cycle system for reverse osmosis desalination," *Desalination*, vol. 183, pp. 73-80, Nov. 2005.
- [32] D. Manolakos *et al.*, "Experimental evaluation of an autonomous low-temperature solar Rankine cycle system for reverse osmosis desalination," *Desalination*, vol. 203, pp. 366-374, Feb. 2007.
- [33] A. S. Nafey and M. A. Sharaf, "Combined solar organic Rankine cycle with reverse osmosis desalination process: Energy, exergy, and cost evaluations," *Renewable Energy*, vol. 35, pp. 2571-2580, Nov. 2010.
- [34] J. G. Haidar and J. I. Ghojel, "Waste heat recovery from the exhaust of low-power diesel engine using thermoelectric generators," 2001, pp. 413-418.
- [35] N. Espinosa *et al.*, "Modeling a Thermoelectric Generator Applied to Diesel Automotive Heat Recovery," *Journal of Electronic Materials*, vol. 39, pp. 1446-1455, Jun. 2010.
- [36] C.-T. Hsu *et al.*, "Experiments and simulations on low-temperature waste heat harvesting system by thermoelectric power generators," *Applied Energy*, vol. 88, pp. 1291-1297, Apr. 2011.
- [37] X. Gou *et al.*, "Modeling, experimental study and optimization on low-temperature waste heat thermoelectric generator system," *Applied Energy*, vol. 87, pp. 3131-3136, Oct. 2010.

- [38] E. W. Miller *et al.*, "Modeling Energy Recovery Using Thermoelectric Conversion Integrated with an Organic Rankine Bottoming Cycle," *Journal of Electronic Materials*, vol. 38, pp. 1206-1213, Mar. 2009.
- [39] E. W. Miller, "Integrated Dual Cycle Energy Recovery Using Thermoelectric Conversion and an Organic Rankine Bottoming Cycle," M.S. Thesis, Dept. Mech. Eng., Oregon State Univ., Corvallis, 2010.
- [40] S. Noie-Baghban and G. Majideian, "Waste heat recovery using heat pipe heat exchanger (HPHE) for surgery rooms in hospitals," *Applied Thermal Engineering*, vol. 20, pp. 1271-1282, Oct. 2000.
- [41] F. Yang *et al.*, "Waste heat recovery using heat pipe heat exchanger for heating automobile using exhaust gas," *Applied Thermal Engineering*, vol. 23, pp. 367-372, Feb. 2003.
- [42] M. A. Abd El-Baky and M. M. Mohamed, "Heat pipe heat exchanger for heat recovery in air conditioning," *Applied Thermal Engineering*, vol. 27, pp. 795-801, Mar. 2007.
- [43] R. K. Shah and D. P. Sekulic, *Fundamentals of Heat Exchanger Design*. Hoboken, NJ :: John Wiley & Sons,, 2003.
- [44] V. Pandiyarajan *et al.*, "Experimental investigation on heat recovery from diesel engine exhaust using finned shell and tube heat exchanger and thermal storage system," *Applied Energy*, vol. 88, pp. 77-87, Jan. 2011.
- [45] S. Mavridou *et al.*, "Comparative design study of a diesel exhaust gas heat exchanger for truck applications with conventional and state of the art heat transfer enhancements," *Applied Thermal Engineering*, vol. 30, pp. 935-947, Jun. 2010.
- [46] D. B. Tuckerman and R. F. W. Pease, "High-Performance Heat Sinking for VLSI," *IEEE Electron Device Letters*, vol. 2, no. 5, pp. 126-129, May 1981.
- [47] F. Incropera *et al.*, *Fundamentals of Heat and Mass Transfer*, 6th ed. Hoboken NJ: John Wiley, 2007.
- [48] "Photo Chemical Machining (PCM) - An Overview," Qualitetch Components Ltd.
- [49] D. M. Allen, "Photochemical Machining: From Manufacturing's Best Kept Secret to a \$6 Billion per annum, Rapid Manufacturing Process." CIRP Journal of Manufacturing Systems, 2005.
- [50] D. Jacobson and G. Humpston, *Principles of brazing*. Materials Park NY: ASM International, 2005.

- [51] G. L. Morini *et al.*, “A critical review of the measurement techniques for the analysis of gas microflows through microchannels,” *Experimental Thermal and Fluid Science*, vol. 35, pp. 849-865, Sep. 2011.
- [52] C. Choi *et al.*, “Surface wettability effect on flow pattern and pressure drop in adiabatic two-phase flows in rectangular microchannels with T-junction mixer,” *Experimental Thermal and Fluid Science*, vol. 35, pp. 1086-1096, Sep. 2011.
- [53] K. Yeong *et al.*, “Characterisation of liquid film in a microstructured falling film reactor using laser scanning confocal microscopy,” *Experimental Thermal and Fluid Science*, vol. 30, pp. 463-472, May 2006.
- [54] *MATLAB*. Natick, MA: MathWorks, 2009.
- [55] C. Ward, “Design and Performance of a Small Scale Waste Heat Recovery Unit,” M.S. Thesis, Dept. Mech. Eng., Oregon State University, Corvallis, 2011.
- [56] S. A. Klein, *Engineering Equation Solver (EES)*. Madison, WI: F-Chart Software, 2011.
- [57] J. D’Errico, *inpaint_nans*. 2006.
- [58] R. Figliola and D. Beasley, *Theory and Design for Mechanical Measurements*, 2nd ed. New York: Wiley, 1995.

Appendices

Appendix A Additional Results

A.1 Additional Microscope Image Processing Results

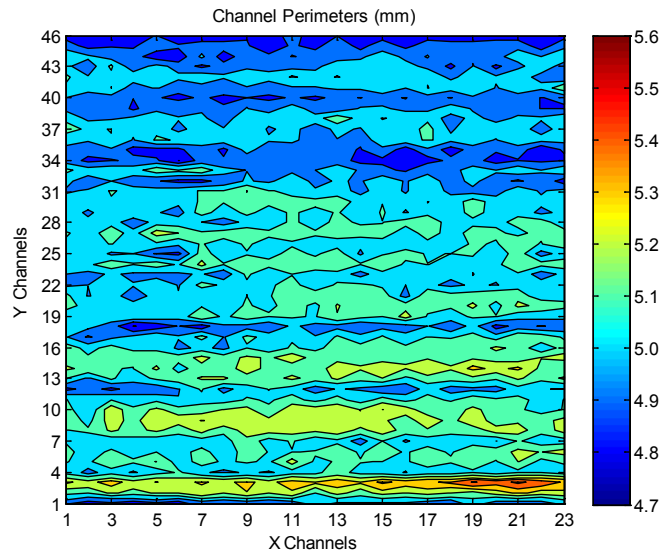


Figure A.1. Air channel perimeters from the Microscope Image Processing

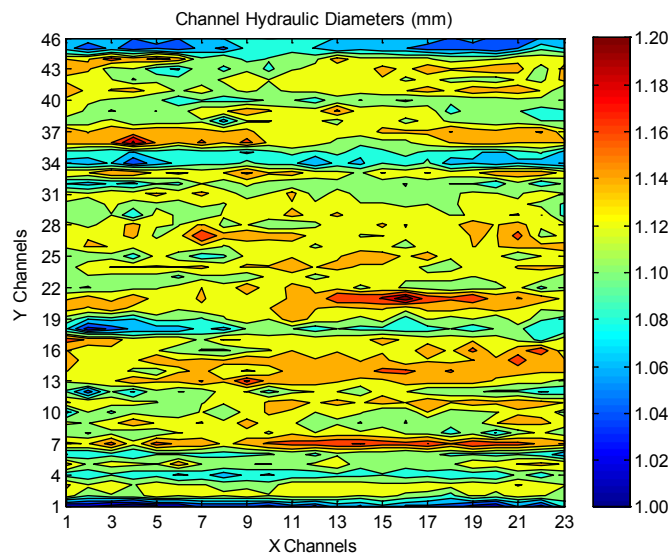


Figure A.2. Air channel hydraulic diameters from the Microscope Image Processing

A.2 Additional Shim Profile Analysis Results

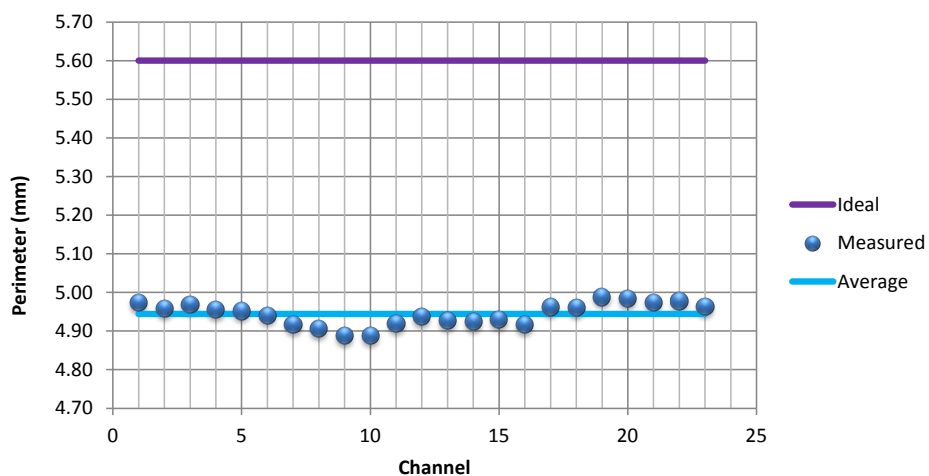


Figure A.3. Air shim perimeters

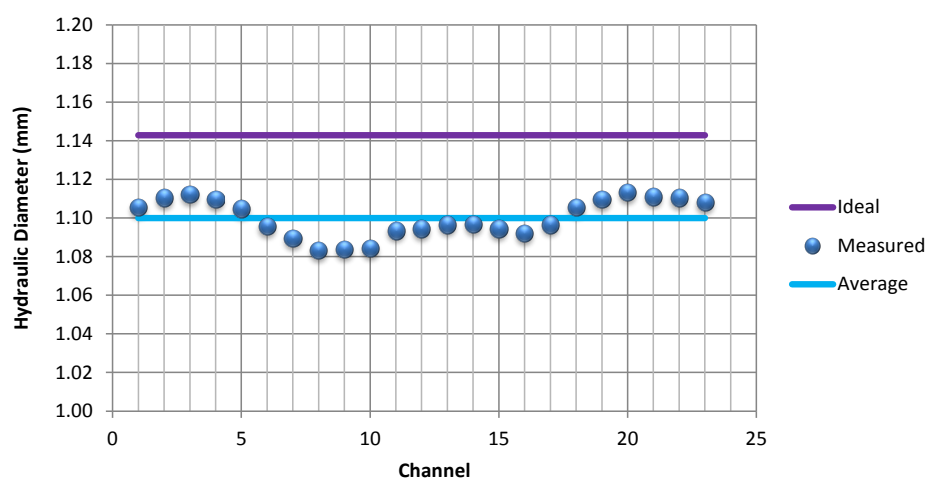


Figure A.4. Air shim hydraulic diameters

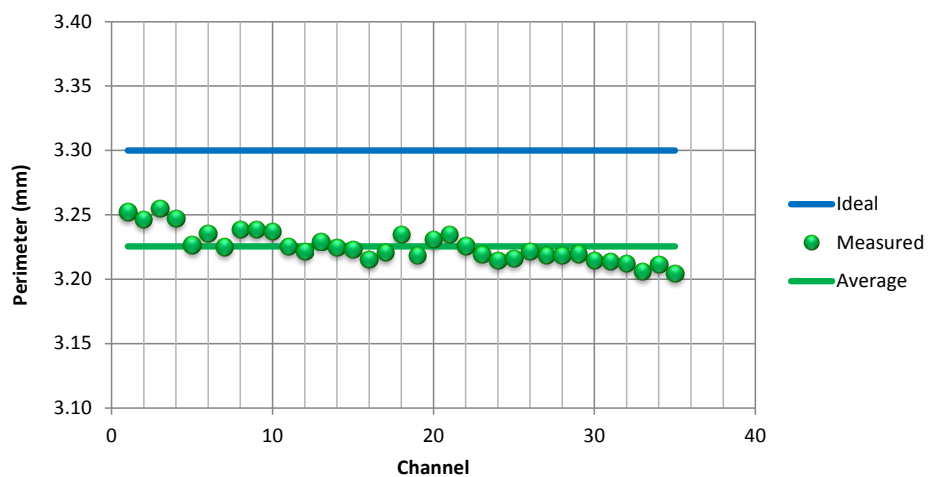


Figure A.5. Oil shim perimeter results

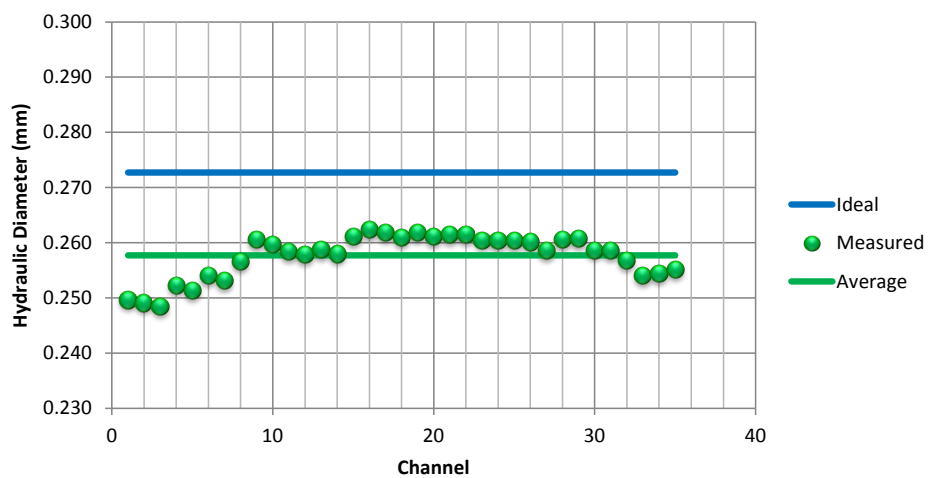


Figure A.6. Oil shim hydraulic diameter results

A.3 Cold Oil Thermal Test Results

In this appendix, the performance measurements of the cold oil inlet (53 °C average) cases are presented.

A.3.1 Duty

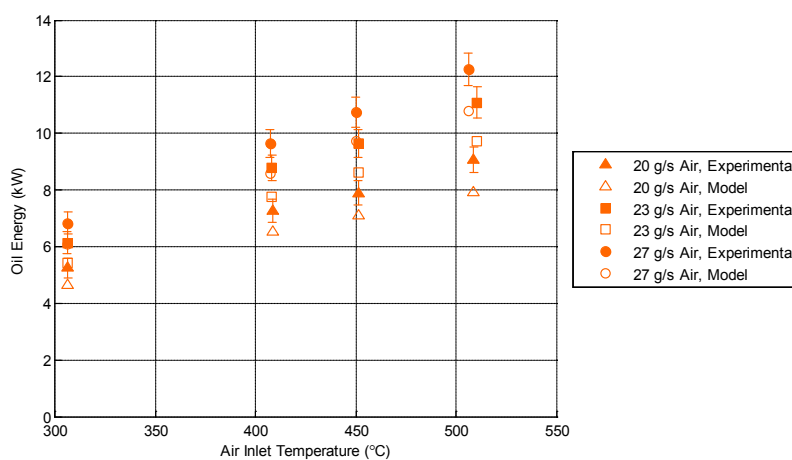


Figure A.7. Duty for 40 g/s cold oil flow

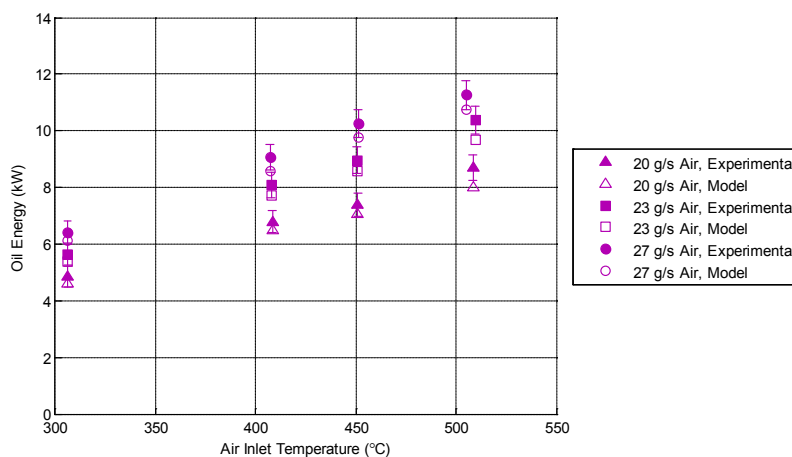


Figure A.8. Duty for 47 g/s cold oil flow

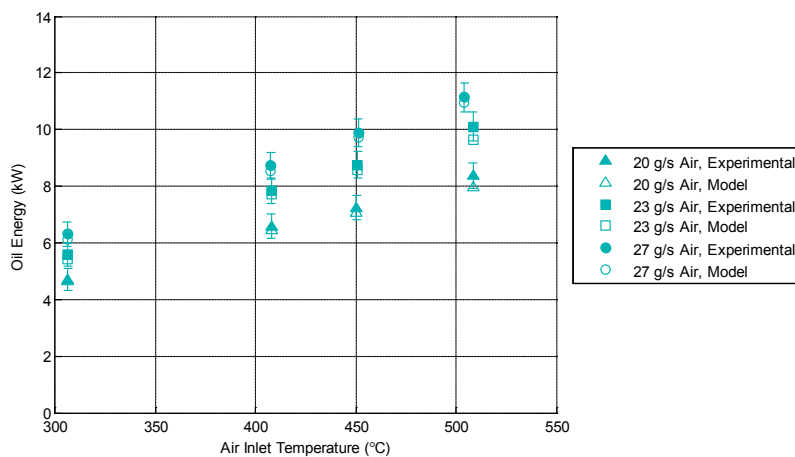


Figure A.9. Duty for 55 g/s cold oil flow

A.3.2 Effectiveness

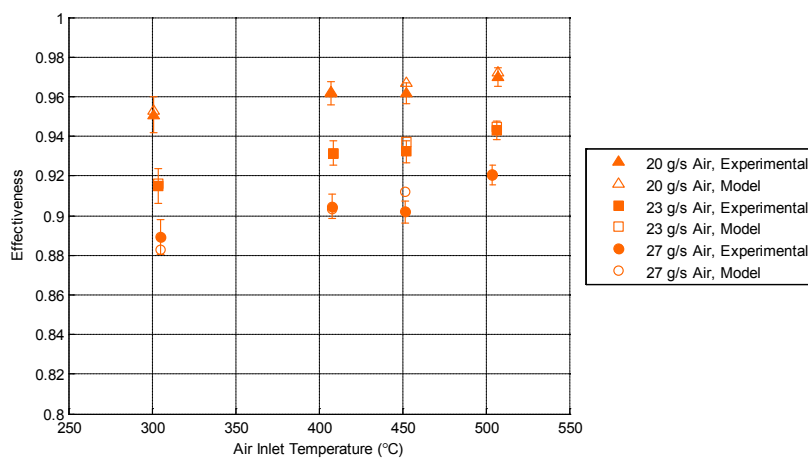


Figure A.10. Effectiveness for 40 g/s cold oil flow

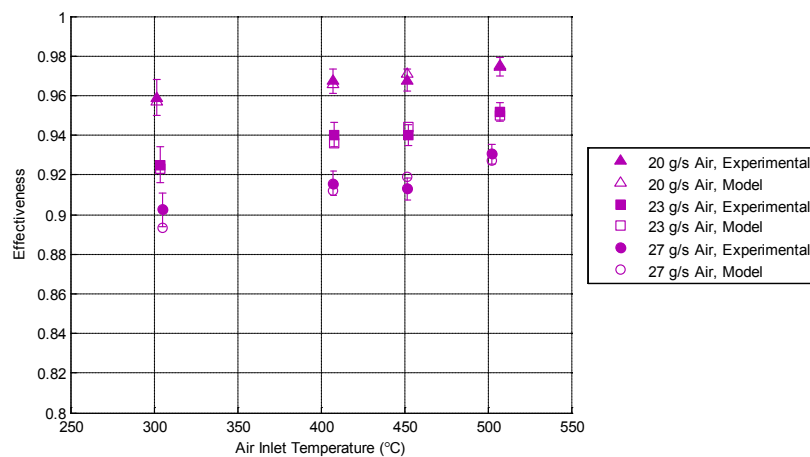


Figure A.11. Effectiveness for 47 g/s cold oil flow

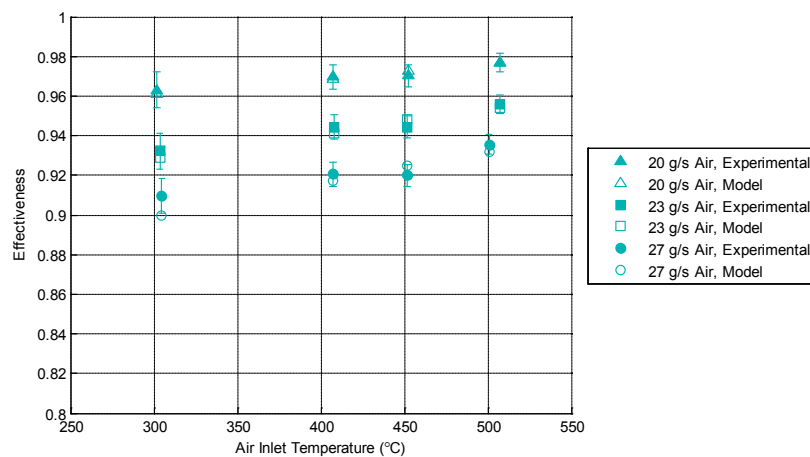


Figure A.12. Effectiveness for 55 g/s cold oil flow

A.3.3 Air Pressure Drop

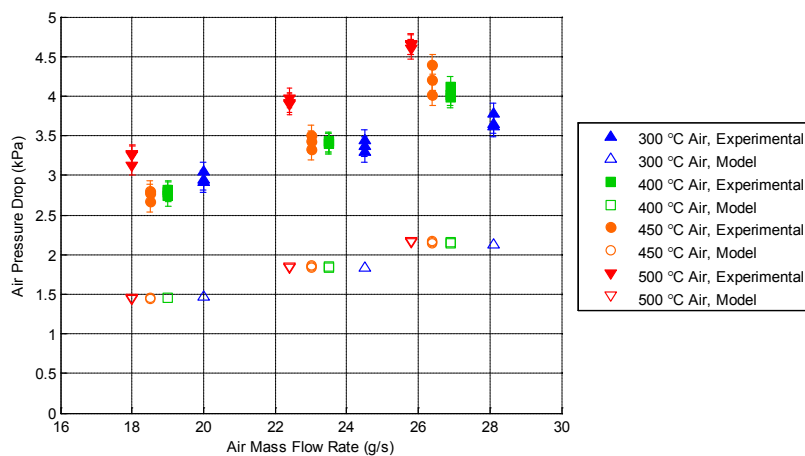


Figure A.13. Air pressure drop for cold oil

A.3.4 Oil Pressure Drop

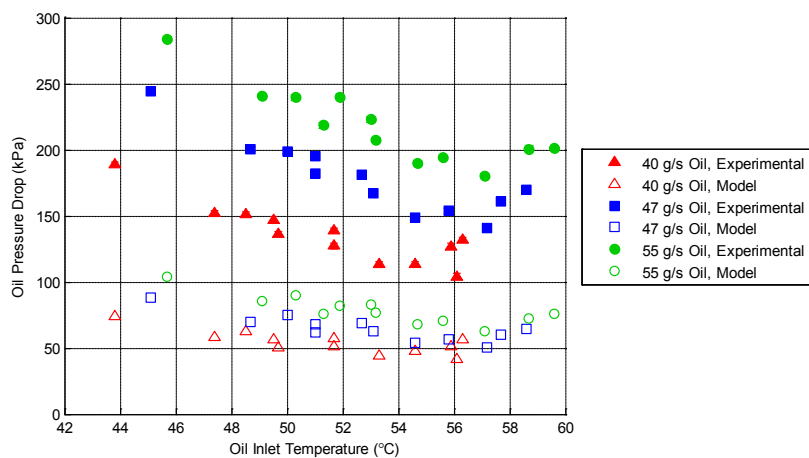


Figure A.14. Oil pressure drop for cold oil

A.4 Additional Oil Inlet Temperature Comparisons

A.4.1 Duty Comparison

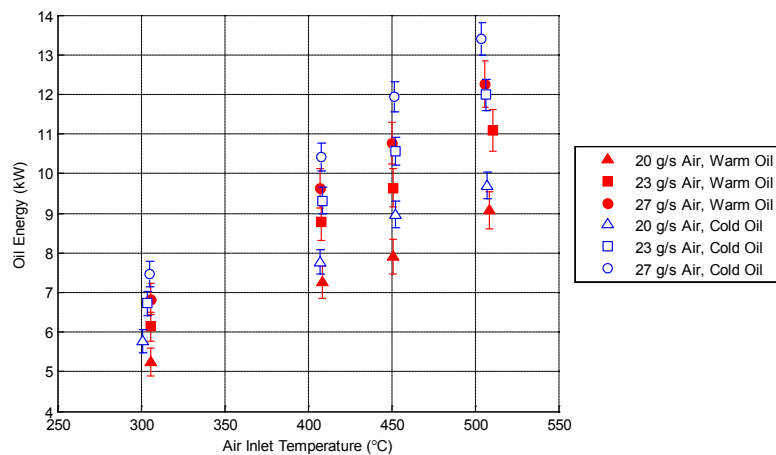


Figure A.15. Duty comparison for 40 g/s oil

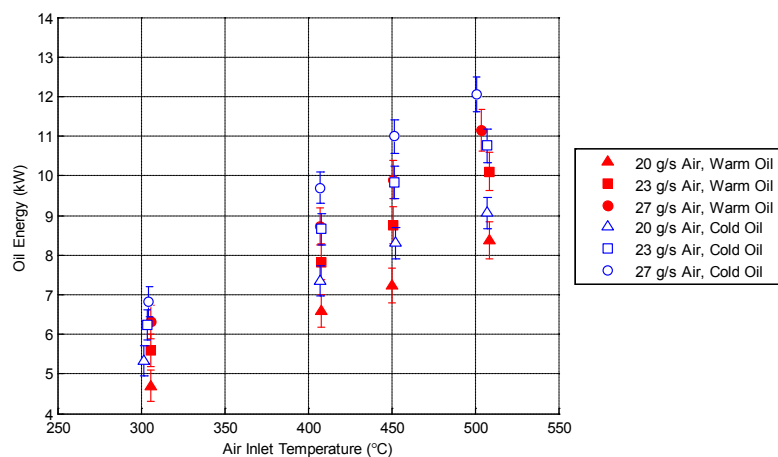
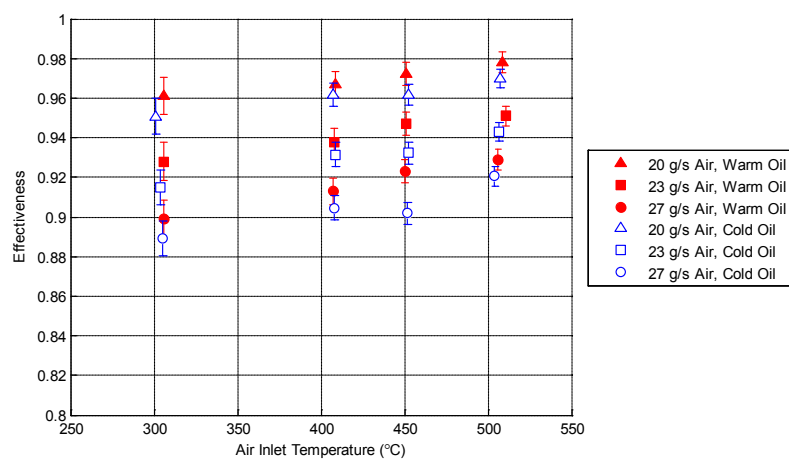
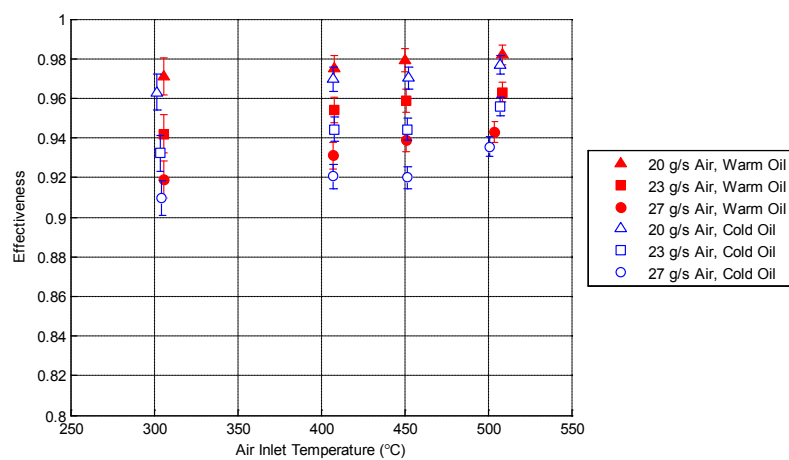


Figure A.16. Duty comparison for 55 g/s oil

A.4.2 Effectiveness Comparison

**Figure A.17. Effectiveness comparison 40 g/s oil****Figure A.18. Effectiveness 55 g/s oil**

A.4.3 Oil Pressure Drop Comparison

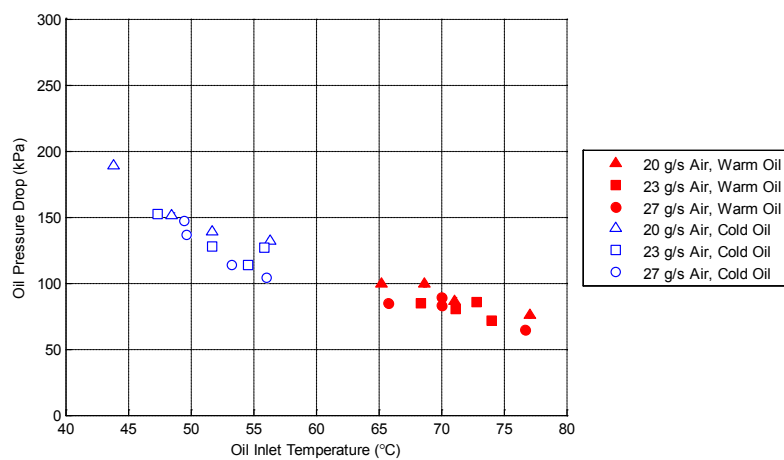


Figure A.19. Oil pressure drop comparison for 40 g/s oil flow

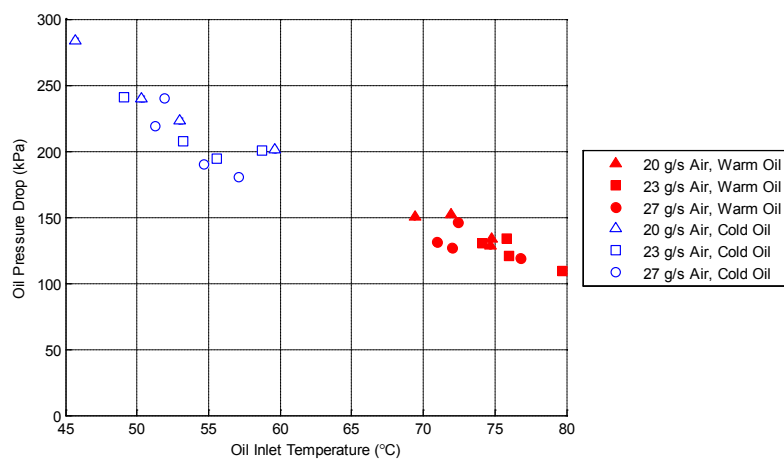


Figure A.20. Oil pressure drop comparison for 55 g/s oil flow

Appendix B Maximum Velocity Profiles

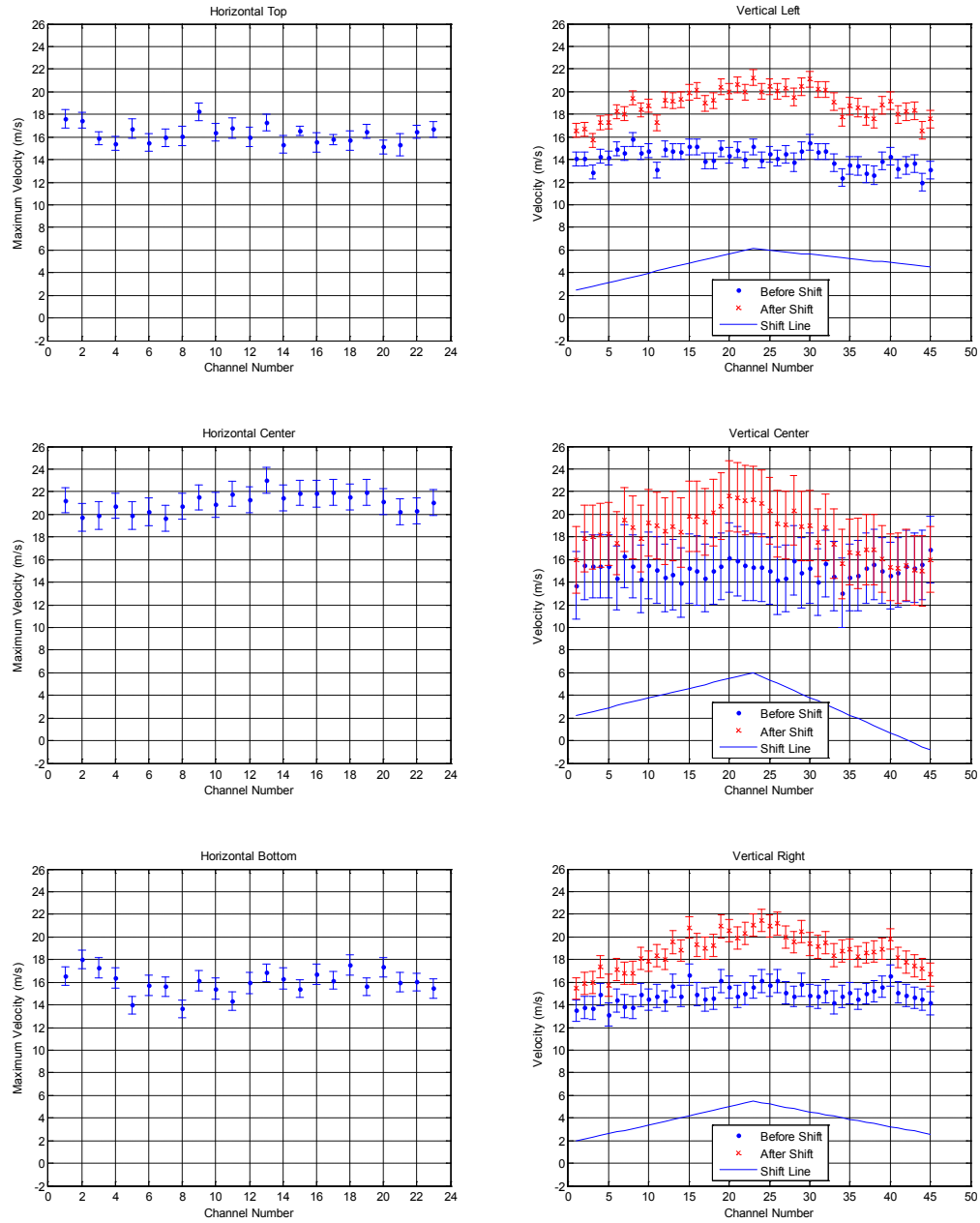


Figure B.1. Extracted maximum channel velocities: horizontal scans on the left and vertical scans with shifts on the right

Appendix C MATLAB Code

Relevant MATLAB programs are presented here.

C.1 Microscope Image Analysis

C.1.1 microscope_image_processing.m

```
% Microscope Image Processing Program
% Created by: James Yih
% Created on: 2-14-2011
% This program analyzes microscope images of the Heat Recovery Unit
air
% channels. The purpose of this program is to measure the channel
area and
% perhaps the overall dimensions of all 1000+ channels.

% Dependent on function files:
% imoverlay.m, createLine.m, createArc.m, and sortRegions.m.

%% Initialize
clear, clc
close all
imgFolder = ['C:\Users\James\Working Home\Graduate Work\',...
    'Heat Recovery Unit\Image Processing\Images\',...
    'Inlet x10 Set 2 2-20-2011\'];

name1 = 1:192;
name2 = 1:2:23;
name3 = 1:3:46;

imgExt = '.jpg';

% Number of rows and columns of channels
nr = 46;
nc = 23;
% Channel areas
Areas = zeros(nr,nc);
Perimeters = zeros(nr,nc);

% Photo information
np = 12;    % number of pictures in a complete row of channels
ncp = 2;    % number of columns in a picture
nrp = 3;    % number of rows in a picture

%% Area Conversion Factor
x_scale = 5.8/572;    % mm/pixel in the x direction
```

```

y_scale = 3.9/386;      % mm/pixel in the y direction
AF = x_scale*y_scale;   % Area per pixel, mm^2/pixel

%% Analyze Images
J = 1; K = 1;    % Loop counters for the image names
tic
for I = 1:length(name1)
    n1 = sprintf('%03.0f',name1(I));
    n2 = sprintf('%02.0f',name2(J));
    n3 = sprintf('%02.0f',name3(K));
    imgName = [n1,'-',n2,'-',n3];
    %imgName = '179-21-43';

    myImage = imread([imgFolder,imgName,imgExt]);
    figure, imshow(myImage), set(gcf,'Name','Original Image')

    %% Load Gray
    grayimg = rgb2gray(myImage);
    cgrayimg = imadjust(grayimg);    % adjust contrast
    figure, imshow(grayimg), set(gcf,'Name','Gray Image')
    figure, imshow(cgrayimg), set(gcf,'Name','Gray Image After
Contrast Adjustment')

    %% Black and White
    thresh = graythresh(cgrayimg);
    BWimg = im2bw(cgrayimg,thresh);
    figure, imshow(BWimg), set(gcf,'Name','Black and White Image')

    %% Dilate
    seHL = strel('line', 3, 0);    % Horizontal line structural
element
    seVL = strel('line', 3, 90);   % Vertical line structural
element
    seHV = [seHL, seVL];          % Combined structural element
    BWdil = imdilate(BWimg, seHV);
    figure, imshow(BWdil), set(gcf,'Name','Dilated Image')

    %% Erode
    %    BWero = imerode(BWdil, seHV);
    %    figure, imshow(BWero)

    %% Canny Edge
    % The Canny method is good according to the MATLAB help on the
edge
    % function.
    channels = edge(BWdil,'canny');
    channels = bwmorph(channels,'bridge');
    figure, imshow(channels), set(gcf,'Name','Canny Edge Detection')

    %figure, imshow(imoverlay(myImage,channels,[1,1,0]))

```

```

%title('Minimally Processed Edge Detection Overlay')

%% Remove Channels on the Border
% To remove channels on the border, draw a box along the border
that
% will connect any channels almost touching the border.

% Draw the box
sc = size(channels);
lw = 10; % Line Width
channels(1:sc(1),1:lw) = 1;
channels(1:sc(1),(sc(2)-lw):sc(2)) = 1;
channels(1:lw,1:sc(2)) = 1;
channels((sc(1)-lw):sc(1),1:sc(2)) = 1;
figure, imshow(channels), set(gcf,'Name','Border Box')

% Remove channels along border
channels = imclearborder(channels,8);
figure, imshow(channels), set(gcf,'Name','Border Channels
Removed')

%% Remove Small Areas
channels = bwareaopen(channels,400);
figure, imshow(channels), set(gcf,'Name','Small Particles
Removed')

%% Edge Overlay
%figure, imshow(imoverlay(myImage,channels,[0,1,1]))
%title('Somewhat Processed Edge Detection Overlay')

%% Fill
fillchan = imfill(channels,'holes');
figure, imshow(fillchan), set(gcf,'Name','Filled Channels')

%% Redilate
seD = strel('diamond',1);
dilchan = imdilate(fillchan,seD);
dilchan = imdilate(dilchan,seD);
figure, imshow(dilchan), set(gcf,'Name','Smoothered Channels')

% Create the channel top line and bottom arc
[linemask, LL, LE] = create_line(dilchan,x_scale,y_scale);
[arcmask, AL, AE] = create_arc(dilchan,8,x_scale,y_scale);
% LL = Line Lengths, LE = Line Ends
% AL = Arc Lengths, AE = Arc Ends

%% Show line and arc masks
showmask = imoverlay(dilchan,linemask,[0,.75,1]);
showmask = imoverlay(showmask,arcmask,[0,1,0]);
figure, imshow(showmask), set(gcf,'Name','Line and Arc Masks')

```

```

%% Overlay masks and clean the image
chanmask = imoverlay(dilchan,linemask,[1,1,1]);
chanmask = imoverlay(chanmask,arcmask,[1,1,1]);
chanmask = im2bw(chanmask);
chanmask = bwmorph(chanmask,'bridge');
chanmask = imfill(chanmask,'holes');
figure, imshow(chanmask), set(gcf,'Name','Incorporate Masks and
Fill Regions')

% Remove small particles (from arc mask)
chanmask = bwareaopen(chanmask,300);
% Clean up with dilate, bridge, fill, erode
chanmask = imdilate(chanmask,seHV);
chanmask = bwmorph(chanmask,'bridge');
chanmask = imfill(chanmask,'holes');
chanmask = imerode(chanmask,seHV);
figure, imshow(chanmask), set(gcf,'Name','Clean Up with Dilate,
Bridge, Fill, and Erode')

%% Draw the boundaries (via edge detection)
edgemask = edge(chanmask,'canny');
edgemask = bwmorph(edgemask,'bridge');
% figure, imshow(imoverlay(myImage,edgemask,[0,1,1])),
set(gcf,'Name','Edge Boundary')

%% Plot final edge overlay
h = figure;
imshow(imoverlay(myImage,edgemask,[0,1,0]))
set(gcf,'Name','Final Image')
% Remove title and axis (for image saving)
%set(gca,'position',[0,0,1,1],'visible','off')

%% Fill the edgemask for region processing
edgemask = imfill(edgemask,'holes');

%% Acquire region information
regioninfo = regionprops(edgemask,'all');
N = size(regioninfo,1); % number of regions

%% Sort the region information based on centroid
regioninfo = sort_regions(regioninfo);

%% Calculate and Plot Area Information
hold on
pr = floor((I-1)/np)+1; % photo row
pc = mod(I-1,np)+1; % photo column

for i = 1:N

```

```

    % The second part of the if statement is required for the
last row
    % which only has one row of channels.
    if I <= 180
        ar = pr*nrp - (mod(i+2,nrp) - 1) - 1;    % actual row
        ac = pc*ncp + floor((i-1)/nrp) - 1;    % actual column
    else
        ar = 46;
        ac = pc*ncp + i - 2;
    end
    CN = nc*(ar - 1) + ac;    % Channel number
    % Area
    Areas(ar,ac) = AF*regioninfo(i).Area;
    % Perimeter
    % Recall LL = Line Lengths, LE = Line Ends,
    %         AL = Arc Lengths, AE = Arc Ends
    % LE and AE => 1,2 = x1 x2; 3,4 = y1 y2
    % LAO = Line-Arc Offset
    % The Line-Arc Offset accounts for the space between the line
    % end points and the arc end points
    LAO_left = sqrt((LE(1,i)-AE(1,i))^2 + (LE(2,i)-AE(2,i))^2);
    LAO_right = sqrt((LE(3,i)-AE(3,i))^2 + (LE(4,i)-AE(4,i))^2);

    Perimeters(ar,ac) = LL(i)+AL(i)+LAO_left+LAO_right;

    str = [num2str(CN),': ',num2str(Areas(ar,ac),3), ' mm^2'];
    coord = regioninfo(i).Centroid;
    text(coord(1)-33,coord(2),str,'color',[1,1,0])
end
hold off

%% Save Image Files with Overlay and Area Calculation
%   saveFolder = ['C:\Users\James\Working Home\Graduate Work\',...
%   'Heat Recovery Unit\Image Processing\Images\Inlet x10 Set 2
Processed\JPEG Process 2\'];
%   print(h,'-djpeg',[saveFolder,imgName]) % Save the figure
%   close(h) % Close the figure

%% Increment name counters
if J == length(name2)
    J = 1;
    K = K + 1;
else
    J = J + 1;
end
clc
fprintf('Processing %3.0f / %3.0f \n',I,length(name1))
end
% Calculate the hydraulic diameters
D_h = 4*Areas./Perimeters;
fprintf('Finished!\n')

```



```
% Write Areas, Perimeters, and Hydraulic Diameters to a .mat file
save('Image Processing Results.mat', 'Areas', 'Perimeters', 'D_h')

toc

% fin!
```

C.1.2 create_line.m

```

function [BW_line_mask_out,line_lengths,line_ends] =
create_line(BW_image_in,x_scale,y_scale)
% createLine creates a smooth line using a polynomial fit that
represents
% the top of each channel. This function assumes that the black and
white
% input image (of type logical) has been preprocessed to isolate the
% channels.
%
% x_scale and y_scale are the pixel to length conversion factors
necessary
% for calculating the length of each line.
%
% The output is a line mask, the length of each line (in physical
units),
% and the (x,y) coordinates of the ends of each line (in physical
units).
%
% Dependencies: sort_regions.m
%
% Updated 10/31/11, James Yih

% Initialize the linemask
linemask = zeros(size(BW_image_in));

% Get properties of each region
bprops = regionprops(BW_image_in,'all');
N = size(bprops,1); % Number of regions
L = size(1,N); % Line lengths
line_ends = zeros(4,N); % Two points for each region/line
% row 1 = x1, row 2 = y1
% row 3 = x2, row 4 = y2

% Determine the boundaries of each region
boundaries = bwboundaries(BW_image_in);
for i = 1:N
    B = boundaries{i};

    % Right top point (x,y)
    rt = [bprops(i).Extrema(3,1),bprops(i).Extrema(3,2)];
    % Extrema are calculated at corners. Round appropriately to
determine
    % the pixel coordinate.
    rt(1) = floor(rt(1));
    rt(2) = ceil(rt(2));

    % To find nearest matching pixel, find all matching X and Y then
add

```

```

    % the vectors. If there is a 2 (hopefully in only one spot
because of
    % boundary function), then the X,Y pair is a match of the Right-
Top
    % point.
    compareX = B(:,2) == rt(1);
    compareY = B(:,1) == rt(2);
    compareXY = compareX+compareY;
    [~, rtloc] = max(compareXY);
    Btop = B(1:rtloc,:);    % Top line points

    % Fit a line
    % Possibly move the line up??
    P = polyfit(Btop(:,2),Btop(:,1),1);
    px = linspace(Btop(1,2),Btop(end,2),Btop(end,2)-Btop(1,2)+1);
    % B(end,1)-B(1,2)+1, without the +1 there is a hole in the center
    py = polyval(P,px);
    % Rasterize the line
    pp = [round(px);round(py)];

    % Apply the line to the mask
    for i2 = 1:length(px)
        linemask(pp(2,i2),pp(1,i2)) = 1;
    end

    % Calculate line length (for perimeter calculation)
    %      sqrt((x1 - x2)^2      + (y1 - y2)^2)
    L(i) = sqrt(((pp(1,1)-pp(1,end))*x_scale)^2 + ((pp(2,1)-
pp(2,end))*y_scale)^2);

    % Output the ends of the line
    line_ends([1,3],i) = [pp(1,1); pp(1,end)]*x_scale;
    line_ends([2,4],i) = [pp(2,1); pp(2,end)]*y_scale;

end
% Convert the linemask into a logical array before sending it out.
BW_line_mask_out = logical(linemask);
% Sort the regions so the output is in a predictable order
[~,Li] = sort_regions(bprops);
% Apply the sort to the lengths
line_lengths = L(Li);

% fin!

```

C.1.3 create_arc.m

```

function [BW_arc_mask_out,arc_lengths,arc_ends] =
create_arc(BW_image_in,pn,x_scale,y_scale)
% createArc takes in a black and white image of the channels then
draws the
% bottom arc. This function assumes that the image has been
preprocessed
% and the channels are isolated.
%
% Inputs:
% BW_image_in should be a 2D logical array.
% pn is the order of the polynomial fit.
% x_scale and y_scale are the pixel to length conversion factors
necessary
% for calculating the length of each created line.
%
% Outputs:
% BW_arc_mask_out is a 2D logical array the same size as BW_image_in.
% arc_lengths is a vector of the arc lengths
% arc_ends is a vector of two pairs of x and y coordinates that are
the end
% points of the arc for each region
%
% Dependencies: Central_Difference.m, sort_regions.m
%
% Updated 10/31/11, James Yih

% Initialize arc mask
arcmask = zeros(size(BW_image_in));

% Get properties of each region
bprops = regionprops(BW_image_in,'all');
N = size(bprops,1);      % Number of regions
al = zeros(1,N);        % arc lengths variable
arc_ends = zeros(4,N);   % arc end points
                        % row 1 = x1, row 2 = y1
                        % row 3 = x2, row 4 = y2

% Determine the boundaries of each region
boundaries = bwboundaries(BW_image_in);

% Polynomial order
%pn = 8;

for i = 1:N
    B = boundaries{i};

    % Right top point (x,y)
    rt = bprops(i).Extrema(3,:);

```

```

    % Extrema are calculated at corners. Round appropriately to
determine
    % the pixel coordinate.
    rt(1) = floor(rt(1));
    rt(2) = ceil(rt(2));

    % To find nearest matching pixel, find all matching X and Y then
add
    % the vectors. If there is a 2 (hopefully in only one spot
because of
    % boundary function), then the X,Y pair is a match of the Right-
Top
    % point.
    compareX = B(:,2) == rt(1);
    compareY = B(:,1) == rt(2);
    compareXY = compareX+compareY;
    [~, rtloc] = max(compareXY);

    % Polynomial fit with entire bottom section
    BBx = B(rtloc:end,2);      % Bottom Boundary x
    BBy = B(rtloc:end,1);      % Bottom Boundary y
    % Center and scale
    meanBBx = mean(BBx);
    stdBBx = std(BBx);
    BBx_bar = (BBx-meanBBx)/stdBBx;
    % Poly fit
    pArc = polyfit(BBx_bar,BBy,pn);

    rtx = B(rtloc,2);
    ltx = B(end,2);      % lt = left-top

    arcx = linspace(ltx,rtx,rtx-ltx+1);
    arcx_bar = (arcx-meanBBx)/stdBBx;
    % Must use x_bar for evaluating the polynomial
    arcy = polyval(pArc,arcx_bar);
    arc = [round(arcx);round(arcy)];

    % Apply the arc to the mask
    for i4 = 1:length(arcx)
        arcmask(arc(2,i4),arc(1,i4)) = 1;
    end

    % Calculate arc lengths (for perimeter calculation)
    % First, take the derivative
    arc_der = Central_Difference(arc(1,:)*x_scale,arc(2,:)*y_scale);
    % Compute the integrand
    I = sqrt(1+arc_der.^2);
    % Integrate
    al(i) = trapz(arcx*x_scale,I);

```

```

    % Output arc ends
    arc_ends([1,3],i) = [arc(1,1);arc(1,end)]*x_scale;
    arc_ends([2,4],i) = [arc(2,1);arc(2,end)]*y_scale;

end
% Convert the arcmask into a logical array -> BW image.
arcmask = logical(arcmask);
% Bridge the arc as best as possible
arcmask = bwmorph(arcmask,'bridge');
% Send the arcmask out
BW_arc_mask_out = arcmask;

% Sort regions
[~,al_i] = sort_regions(bprops);
arc_lengths = al(al_i);

% fin!

```

C.1.4 sort_regions.m

```

function [regions_out,sorted_indices] = sort_regions(regions_in)
% This function sorts the regions in the image based on the location
of
% their centroid. The first region starts at the top left corner
(lowest x,
% lowest y centroid). Regions increase going down then to the right.
% i.e. [ 1  4
%       2  5
%       3  6 ]
% This numbering scheme was chosen because it is nearly the same
numbering
% scheme as the regionprops command. However, since the regionprops
% numbering scheme is inconsistent, this function file is necessary.
% The input, regions_in, contains all the information about the
regions (a
% result of regionprops(imgName,'all')).
% The output are regions_out, containing all of the information,
sorted and
% i_sorted, the sorted indices.

n = size(regions_in,1);    % number of regions
%% Assign the centroids
for i = 1:n
    C(i,:) = regions_in(i).Centroid;
end
% Add a counter vector to the front to keep track of original
locations.
% Round the numbers to make it look nice (and make it easier later to
use
% the location indices).
C = round([1:n]',C));
B = zeros(size(C));

%% Examine the possibilities based on the number of regions
if n == 6
    % n = 6, maximum number of regions present
    % Sort by the X coordinate first
    C = sortrows(C,2);
    % Next sort by the Y coordinate in threes
    for i = 1:2
        a = 3*i-2;
        b = 3*i;
        B(a:b,:) = sortrows(C(a:b,:),3);
    end
elseif n == 3
    % n = 3, side image with 3 vertical regions
    % The only step required here is to sort by the Y coordinate
    B = sortrows(C,3);
elseif n == 2

```

```
% n = 2, top image with 2 horizontal regions
% Sort by the X coordinate
B = sortrows(C,2);
else
    % n = 1, no sorting required
    B = C;
end
%% Assign the output
% The indices in column 1 of matrix B are used to assign the output
regions_out = regions_in(B(:,1),1);
sorted_indices = B(:,1);

% fin!
```


C.2 Shim Profile Analysis

C.2.1 profile_area_calculator.m

```
% This program uses profile data collected from the ZeScope
profilometer to
% calculate channel cross-sectional areas and hydraulic diameters.
% The data is read through .mat files
% The data is assumed to be "nice" in that it doesn't have much bow
and
% there isn't a lot of noise.
% Created: 5/2/2011
% James Yih

clear, clc, close all
tic
%% File Directories
filedir = 'C:\Users\James\Working Home\Graduate Work\Heat Recovery
Unit\Profilometer\Shims\Processed Data\mat\';
% filedir = 'C:\Users\James\Working Home\Graduate Work\Heat Recovery
Unit\Profilometer\Shims\Processed Data\mat\Oil Shim 3 Channels\';
savedir = 'C:\Users\James\Working Home\Graduate Work\Heat Recovery
Unit\Profilometer\Shims\Processed Data\Processing Images\';

%% Load Files
% To load a file, provide the filename, tbthresh (top-bottom
threshold),
% the units of the incoming data (um or nm), and the ideal dimensions
of
% the channel (width W and height H).
% Or comment/uncomment the appropriate prewritten file information

%% Oil Shims
% Single Scan
% filename = 'Oil Shim 3 Scan 1.mat';
% % filename = 'Oil Shim 1 2_zmp_prf Original.mat'; % Scan used in
% examples
% tbthresh = 0.6; % threshold for distinguishing tops from bottoms
% units = 'nm';

% Oil stitch
% filename = 'Oil Shim 3 7 scan stitch.mat';
% tbthresh = 0.9; % threshold for distinguishing tops from bottoms
% % units = 'um';

%% Air Shims
% % Single Scan
% filename = 'Air Shim 1 Channel 7.mat';
% tbthresh = 0.9; % threshold for distinguishing tops from bottoms
```

```

% units = 'nm';

% Air Stitch
filename = 'Air Shim 1 6 Scan Stitch.mat';
tbthresh = 0.9;
% units = 'um';

%% Ideal Geometry
% Oil Channel
W = 1.5;      % width, mm
H = 0.15;     % height, mm

% Air Channel
% W = 2;      % width, mm
% H = 0.8;     % height, mm

ideal_area = W*H;          % mm^2
ideal_perimeter = 2*(W+H); % mm
ideal_d = 4*ideal_area/ideal_perimeter; % hydraulic diameter

%% Threshold Parameters
%tbthresh = 0.6; % threshold for distinguishing tops and bottoms
dxthresh = 0.5; % threshold for dx (change in x)
% see Group shim top points and find top edges
zerotol = 0.01; % tolerance for derivatives near zero (as decimal
percentage)
% see Filter Derivatives Near Zero

%% Load the data
load([filedir,filename])
% Data variables loaded: x, y
n = length(x);

%% Convert nm to um
% Can't convert directly to mm because the polyfits will result in
numbers
% so small that MATLAB will round them to zero.
if strcmpi(units,'nm') == 1
    x = x/1000;
    y = y/1000;
end

%% Apply an offset to make all values positive
offset = min(y);
y = y+abs(offset);
y = y-min(y);

%% Plot the data
figure
plot(x,y,'.')

```

```

title(filename(1:end-4),'Interpreter','none')
xlabel('x (\mum)'), ylabel('y (\mum)')
grid on

%% Find the first derivative via central differencing
dy = zeros(n,1);
for i = 1:n
    if i == 1
        % First point, forward differencing
        dy(1) = (y(2)-y(1))/(x(2)-x(1));
    elseif i == n
        % Last point, backward differencing
        dy(n) = (y(n)-y(n-1))/(x(n)-x(n-1));
    else
        % Interior points, central differencing
        dy(i) = (y(i+1)-y(i-1))/(x(i+1)-x(i-1));
    end
end

%% Plot the Derivative
figure
% Original
subplot(2,1,1)
plot(x,y,'. '), hold on
% Top-Bottom Separation Criteria (see Find Shim Tops section)
% plot([x(1),x(end)],tbthresh*max(y)*[1,1],'r')
xlabel('x (\mum)'), ylabel('y (\mum)')
grid on
% Derivative
subplot(2,1,2)
plot(x,dy,'.r')
xlabel('x (\mum)'), ylabel('dy/dx')
grid on

%% Filter derivatives near zero (tops and bottoms of channels)
tol = zerotol*max(abs(dy)); % tolerance close to zero
topnbot = logical(dy <= tol & dy >= -tol); % near zero
dy_0 = dy(topnbot);
x_0 = x(topnbot);
y_0 = y(topnbot);

% Plot the zero tolerance
% subplot(2,1,2), hold on
% plot(x,tol*ones(1,n),'m')
% plot(x,-tol*ones(1,n),'m')

% Plot the filter results
subplot(2,1,1), hold on
plot(x_0,y_0,'.g')
subplot(2,1,2), hold on
plot(x_0,dy_0,'.g')

```

```

%% Find the shim tops
% Shim tops are values greater than the mean
% This condition may need to be altered if the bow is large
tops = logical(y(topnbot) >= tbthresh*max(y));
% x_t and y_t are the coordinates of the tops of the shim
x_t = x_0(tops); y_t = y_0(tops);
% Plot the tops
subplot(2,1,1)
plot(x_t,y_t,'.c')
subplot(2,1,2)
plot(x_0(tops),dy_0(tops),'.c')

%% Group top shim points and find top edges
% Using subtraction and top channel pairing
% (Locate the edges of the tops)
for i = 1:length(x_t)
    if i == length(x_t)
        dx_t(i) = 0;
    else
        dx_t(i) = x_t(i+1)-x_t(i);
    end
end
dx_t = dx_t';

cb = logical(dx_t > dxthresh*max(dx_t)); % channel break, occurs
when the x difference is large
% The condition for picking out large gaps in x is arbitrary and may
need
% to be revisited. (if the aspect ratio of the channel is changed -
is this true?).
cbi = 1:length(dx_t); % channel break index
(relative to dx_t)
cbi = cbi(cb);
num_chan = length(cbi); % number of full channels present
cbi = [cbi, cbi+1]; % add the edge of the opposing channel
feature
cbi = sort(cbi);

% Convert the break index to an index relative to the original
vector, x
% This information will be easier to work with in succeeding parts of
the
% code.
% The function ismember can be used to convert the indices since all
x_t
% will be in the set x and the function is one to one.
% Add the first and last point of x_t for the first and last edges
[tf, edge_loc] = ismember([x_t(cbi);x_t(1);x_t(end)],x);
edge_loc = sort(edge_loc); % this step is necessary for further
processing

```

```

figure
plot(x,y,'. '), hold on
plot(x(edge_loc),y(edge_loc),'ro','MarkerFaceColor','r','MarkerSize',
4)
xlabel('x (\mum)'), ylabel('y (\mum)')
grid on

%% Line drawing on shim tops
% p_tops contains the line coefficients (slope, intercept)
p_tops = zeros(num_chan+1,2);
for i = 1:num_chan+1
    % Keep track of the x values (in this case, the indices) used
    for
        % fitting
        % tfi = top line fit index
        tfi{i} = edge_loc(2*i-1):edge_loc(2*i);
        %tfi{i} = topset{i};
        p_tops(i,:) = polyfit(x(tfi{i}),y(tfi{i}),1);
        shim_top_fit{i} = polyval(p_tops(i,:),x(tfi{i}));

        % Plot the fitted lines
        plot(x(tfi{i}),shim_top_fit{i},'c','LineWidth',2)
    end

%% Line drawing over channels
p_chan_tops = zeros(num_chan,2);
for i = 1:num_chan
    % ctfp = channel top fit points
    x_ctfp{i} = [x(tfi{i}(end)),x(tfi{i+1}(1))];
    y_ctfp{i} = [shim_top_fit{i}(end),shim_top_fit{i+1}(1)];
    p_chan_tops(i,:) = polyfit(x_ctfp{i},y_ctfp{i},1);

    plot(x_ctfp{i},y_ctfp{i},'or')

    % ctf = channel top fit (vector)
    x_ctf{i} = x_ctfp{i};
    y_ctf{i} = polyval(p_chan_tops(i,:),x_ctf{i});

    % Plot the fitted channel tops
    plot(x_ctf{i},y_ctf{i},'m')
end

%% Calculate channel area
for i = 1:num_chan
    a = edge_loc(2*i):edge_loc(2*i+1);
    x_chan = x(a);
    y_chan = y(a);

    line_area = trapz(x_ctf{i},y_ctf{i});

```

```

        inner_area = trapz(x_chan,y_chan);
        area(i) = line_area-inner_area;
    end

    %% Area comparison
    area = area/1000^2;
    area = area';
    %ideal_area = 1.5*0.15; % mm^2
    area_percent = area/ideal_area;
    area_error = 1-area/ideal_area;

    %% Plot areas
    % figure
    % plot(area, '^')
    % grid on

    %% dx_t plot
    % figure
    % plot(dx_t, '.r'), hold on
    % % plot([1,length(dx_t)],mean(dx_t)*ones(1,2),'g')
    % % plot([1,length(dx_t)],dxthresh*max(dx_t)*ones(1,2),'b')
    % xlabel('Top Point #'),ylabel('\Deltax (\num)')
    % grid on

    %% Calculate channel perimeter
    % The channel perimeter is composed of three parts: the top line,
    channel
    % curve, and space between the line and curve.
    % Calculating the perimeter of curve is done by numerical
    differentiation
    % and integration
    P_c = zeros(num_chan,1); % length of curves
    P_t = zeros(num_chan,1); % length of top line
    %P_d = zeros(num_chan,1); % length of the small space between curve
    and top line
    perimeter = zeros(num_chan,1); % perimeter vector
    %gap_check = zeros(num_chan,2);
    for i = 1:num_chan
        % Line length - channel top
        % L = sqrt( (x1-x2)^2 + (y1-y2)^2 )
        P_t(i) = sqrt((x_ctfp{i}(1)-x_ctfp{i}(2))^2+(y_ctfp{i}(1)-
        y_ctfp{i}(2))^2);

        % Channel curve length - channel interior
        % arc length = integral(sqrt(1+(dy/dx)^2))dx
        a = edge_loc(2*i):edge_loc(2*i+1);
        x_chan = x(a);
        y_chan = y(a);
        arc_int = sqrt(1+dy(a).^2);
        P_c(i) = trapz(x_chan,arc_int);
    end

```

```

        % Small space between curve and top line
        %         Left side          +          Right side
%     P_d(i) = y_ctfp{i}(1)-y_chan(1) + y_ctfp{i}(2)-y_chan(end);
%     gap_check(i,1) = y_ctfp{i}(1) > y_chan(1);
%     gap_check(i,2) = y_ctfp{i}(2) > y_chan(end);

    perimeter(i) = P_t(i)+P_c(i);%+P_d(i);
end
perimeter = perimeter/1000;      % convert um to mm

%% Calculate hydraulic diameter
D_h = 4*area./perimeter;

%% Save Figures
% %figurecount = length(findobj('Type','figure'));
% figurename = {'Data';'Surface Separation';'Channel Line';'dx of
Tops'};
% shim_name = 'Oil Shim 1 1';
% for i = 1:4
%     savename = [shim_name,' ',figurename{i}];
%     if i ~= 4
%         set(i,'OuterPosition',[85, 323, 1116, 470]);
%     end
%     print(i,[savedir,savename],'-djpeg')
% end

%% Determine if shim top edges occur above shim top line ends
% cell2mat(y_ctfp) > y(edge_loc(2:end-1))'

toc

% Fin!

```

C.2.2 profile_multi_scan_level_and_stitch.m

```

% This program uses channel profile data collected from the ZeScope
profilometer
% (or any other source) to create a leveled and stitched scan of the
% channels.
% All channels of interest should be bounded by channel walls (kind
of
% obvious?).
% The data is read through .mat files
% Created: 5/9/2011
% James Yih

%% Load Files
clear, clc, close all
filedir = 'C:\Users\James\Working Home\Graduate Work\Heat Recovery
Unit\Profilometer\Shims\Processed Data\mat\';
savedir = 'C:\Users\James\Working Home\Graduate Work\Heat Recovery
Unit\Profilometer\Shims\Processed Data\Processing Images\';

% Oil Shim
% shim_name = 'Oil Shim 3';
% num_scans = 7;
% % ideal_area = 1.5*0.15; % mm^2
% tbthresh = 0.6;

% Air Shim
shim_name = 'Air Shim 1';
num_scans = 6;
ideal_area = 2*0.8; % mm^2
tbthresh = 0.9;

%tbthresh = 0.6; % threshold for distinguishing tops and bottoms
dxthresh = 0.6; % threshold for dx (change in x)
% see Group shim top points and find top edges
zerotol = 0.05; % tolerance for derivatives near zero (as decimal
percentage)
% see Filter Derivatives Near Zero

%% Read all files and process
% Initialize some variables
X = cell(num_scans,1);
Y = cell(num_scans,1);
N = zeros(num_scans,1);
area = cell(num_scans,1);
X_stitch = []; Y_stitch = [];
edge_loc = cell(num_scans,1);
tic
for loop = 1:num_scans
    %% Load the data

```



```

filename = [shim_name, ' Scan ', num2str(loop), '.mat'];
% filename = [shim_name, ' ', num2str(loop), ' Example.mat'];
load([filedir, filename])
% Data variables loaded: x, y
n = length(x);
X{loop} = x; Y{loop} = y;
N(loop) = n;

%% Convert nm to um
% Can't convert directly to mm because the polyfits will result
in numbers
% so small that MATLAB will round them to zero.
x = x/1000;
y = y/1000;

%% Apply an offset to make all values positive
y = y+abs(min(y));
y = y-min(y);

%% Plot the data
% figure
% plot(x,y, '.')
% %title(filename(1:end-4), 'Interpreter', 'none')
% xlabel('x (\mum)'), ylabel('y (\mum)')
% grid on
% %axis equal

%% Find the first derivative via central differencing
dy = zeros(n,1);
for i = 1:n
    if i == 1
        % First point, forward differencing
        dy(1) = (y(2)-y(1))/(x(2)-x(1));
    elseif i == n
        % Last point, backward differencing
        dy(n) = (y(n)-y(n-1))/(x(n)-x(n-1));
    else
        % Interior points, central differencing
        dy(i) = (y(i+1)-y(i-1))/(x(i+1)-x(i-1));
    end
end

%% Plot the Derivative
% figure
% % Original
% subplot(2,1,1)
% plot(x,y, '.'), hold on
% % Top-Bottom Separation Criteria (see Find Shim Tops section)
% plot([x(1), x(end)], tbthresh*max(y)*ones(1,2), 'r')
% xlabel('x (\mum)'), ylabel('y (\mum)')
% grid on

```

```

% title(filename(1:end-4),'Interpreter','none')
% % Derivative
% subplot(2,1,2)
% plot(x,dy,'.r')
% xlabel('x (\mum)'), ylabel('dy/dx')
% grid on

%% Filter derivatives near zero (tops and bottoms of channels)
tol = zerotol*max(abs(dy)); % tolerance close to zero
topnbot = logical(dy <= tol & dy >= -tol); % near zero
dy_0 = dy(topnbot);
x_0 = x(topnbot);
y_0 = y(topnbot);

%% Plot the filter results
% subplot(2,1,1), hold on
% plot(x_0,y_0,'.g')
% subplot(2,1,2), hold on
% plot(x_0,dy_0,'.g')

%% Find the shim tops
% Shim tops are values greater than the mean
% This condition may need to be altered if the bow is large
tops = logical(y(topnbot) >= tbthresh*max(y));
% x_t and y_t are the coordinates of the tops of the shim
x_t = x_0(tops); y_t = y_0(tops);

%% Store top points with respect to the full vector
% tops_loc => indicies of the x vector
% tops_tf => true false = logical vector
[~,tops_loc] = ismember(x_t,x);
tops_tf = zeros(n,1);
tops_tf(tops_loc) = 1;
tops_tf = logical(tops_tf);

%% Plot the tops
% subplot(2,1,1)
% plot(x_t,y_t,'.c')
% subplot(2,1,2)
% plot(x_0(tops),dy_0(tops),'.c')

%% Group top shim points and find top edges
% Using subtraction and top channel pairing
% (Locate the edges of the tops)
dx_t = [];
for i = 1:length(x_t)
    if i == length(x_t)
        dx_t(i) = 0;
    else
        dx_t(i) = x_t(i+1)-x_t(i);
    end
end

```

```

        end
    end
    dx_t = dx_t';

    cb = [];
    cb = logical(dx_t > dxthresh*max(dx_t));    % channel break,
occurs when the x difference is large
    % The condition for picking out large gaps in x is arbitrary and
may need
    % to be revisited.
    cbi = 1:length(dx_t);    % channel break index (relative to
dx_t)
    cbi = cbi(cb);
    num_chan = length(cbi);    % number of full channels present
    cbi = [cbi, cbi+1];    % add the edge of the opposing
channel feature
    cbi = sort(cbi);

    % Convert the break index to an index relative to the original
vector, x
    % This information will be easier to work with in succeeding
parts of the
    % code.
    % The function ismember can be used to convert the indices since
all x_t
    % will be in the set x and the function is one to one.
    % Add the first and last point of x_t for the first and last
edges
    [~, edge_loc{loop}] = ismember([x_t(cbi);x_t(1);x_t(end)],x);
    edge_loc{loop} = sort(edge_loc{loop});    % this step is
necessary for further processing

    %% Plot Edges
    figure
    plot(x,y, '.'), hold on

    plot(x(edge_loc{loop}),y(edge_loc{loop}), 'ro', 'MarkerFaceColor','r', '
MarkerSize',4)
    xlabel('x (\mum)'), ylabel('y (\mum)')
    grid on
    %title(filename(1:end-4), 'Interpreter','none')

    %% Line drawing on shim tops
    % p_tops contains the line coefficients (slope, intercept)
    p_tops = zeros(num_chan+1,2);
    tfi = cell(num_chan+1,1);
    shim_top_fit = cell(num_chan+1,1);
    for i = 1:num_chan+1
        % Keep track of the x values (in this case, the indices)
used for
        % fitting

```

```

    % tfi = top line fit index
    tfi{i} = edge_loc{loop}(2*i-1):edge_loc{loop}(2*i);
    %tfi{i} = topset{i};
    p_tops(i,:) = polyfit(x(tfi{i}),y(tfi{i}),1);
    shim_top_fit{i} = polyval(p_tops(i,:),x(tfi{i}));

    % Plot the fitted lines
    plot(x(tfi{i}),shim_top_fit{i},'c','LineWidth',2)
end

%% For Display: Plot Side Lines
% Left Side

plot(x(1:edge_loc{loop}(1)),polyval(p_tops(1,:),x(1:edge_loc{loop}(1)
)), 'm')
    % Right Side

plot(x(edge_loc{loop}(end):end),polyval(p_tops(end,:),x(edge_loc{loop}
)(end):end)), 'm')

%% Line drawing over channels
p_chan_tops = zeros(num_chan,2);
x_ctfp = cell(num_chan,1);
y_ctfp = cell(num_chan,1);
x_ctf = cell(num_chan,1);
y_ctf = cell(num_chan,1);
for i = 1:num_chan
    % ctfp = channel top fit points
    x_ctfp{i} = [x(tfi{i}(end)),x(tfi{i+1}(1))];
    y_ctfp{i} = [shim_top_fit{i}(end),shim_top_fit{i+1}(1)];
    p_chan_tops(i,:) = polyfit(x_ctfp{i},y_ctfp{i},1);

    % Plot edge points that are used in the calculations
    plot(x_ctfp{i},y_ctfp{i},'or')

    % ctf = channel top fit (vector)
    x_ctf{i} = x_ctfp{i};
    y_ctf{i} = polyval(p_chan_tops(i,:),x_ctf{i});

    % Plot the fitted channel tops
    plot(x_ctf{i},y_ctf{i},'m')
end

%% dx plot
% figure
% plot(dx_t,'.r'), hold on
% plot([1,length(dx_t)],mean(dx_t)*ones(1,2),'g')
% plot([1,length(dx_t)],dxthresh*max(dx_t)*ones(1,2),'b')
% xlabel('Point #'),ylabel('\Delta x (\mum)')
% grid on

```

```

%% Build complete top line across the entire scan
y_top_line = [];
nt = 1;
nc = 1;
% number of channels = num_chan
% number of channel walls = num_chan + 1
% total = 2*num_chan + 1
% loop through the sum of channels and walls

for i = 1:2*num_chan+1
    if i == 1
        % Left end of the scan. shim top + any extra
        xtemp = x(1:edge_loc{loop}(2));
    elseif i == 2*num_chan+1
        % Right end of the scan. shim top + any extra
        xtemp = x(edge_loc{loop}(end-1)+1:n);
    else
        % Scan interior
        xtemp = x(edge_loc{loop}(i)+1:edge_loc{loop}(i+1));
    end

    % Determine which polynomial to use. This is linked to the
loop.
    if mod(i,2) == 1
        % Odd loop number => shim top
        ptemp = p_tops(nt,:);
        nt = nt+1;
    else
        % Even loop number => channel top
        ptemp = p_chan_tops(nc,:);
        nc = nc+1;
    end
    ytemp = polyval(ptemp,xtemp);
    y_top_line = [y_top_line;ytemp];
end
% Plot the combined lines vector
%plot(x,y_top_line,'g-','LineWidth',2)

%% Level top line
ylevel(loop,1) = max(y);
level_line = ylevel(loop)*ones(size(x));
lineshift = level_line-y_top_line;
y_levelled = y+lineshift;

% level point
% level line
% difference
% apply shift

% Plot the level line
plot(x,level_line,'r')

%% Plot leveled scan
figure

```

```

plot(x,y_ leveled, '.'), hold on
grid on
title(filename(1:end-4), 'Interpreter', 'none')
xlabel('x (\mum)'), ylabel('y (\mum)')
ylim([0,180])

%% For Display: Plot Both Leveled Scans
% Plot Scans 1 and 2 on subplots
figure(3)
if loop == 1
    subplot(1,2,1)
    plot(x,y_ leveled, '.')
    title('Oil Shim 3: Example Scan 1')
    xlabel('x (\mum)'), ylabel('y (\mum)')
    grid on
    xlim([-100,5300])
else
    subplot(1,2,2)
    plot(x,y_ leveled, '.r')
    title('Oil Shim 3: Example Scan 2')
    xlabel('x (\mum)'), ylabel('y (\mum)')
    grid on
    xlim([-100,5300])
end

% Plot Scans 1 and 2 on the same plot
if loop == 2
    figure(3), hold on
    plot(x,y_ leveled, '.r')
    title('Oil Shim 3: Example Scans 1 & 2')
    legend('Scan 1', 'Scan 2')
end

%% Stitch Scans
% Determine how many points are at the last shim top (right side
of
% previous scan) and the current shim top (left side of the
current
% scan).
right_length(loop,1) = edge_loc{loop}(end)-edge_loc{loop}(end-
1)+1;
left_length(loop,1) = edge_loc{loop}(2)-edge_loc{loop}(1)+1;

% Test code - Visualize points of interest
%plot(x(edge_loc{loop}(end-
1):edge_loc{loop}(end)),y_ leveled(edge_loc{loop}(end-
1):edge_loc{loop}(end)),'.m')

%plot(x(edge_loc{loop}(1):edge_loc{loop}(2)),y_ leveled(edge_loc{loop}
(1):edge_loc{loop}(2)),'.g')

```

```

if loop == 1
    %% Start the stitch with the first scan
    % Cut off ends that are not full channels
    X_stitch = x(edge_loc{loop}(1):edge_loc{loop}(end));
    Y_stitch = y_levelled(edge_loc{loop}(1):edge_loc{loop}(end));
    N(loop) = length(X_stitch);
    % Shift top edge locations to reflect the cutting
    edge_loc_stitch = edge_loc{loop}-(edge_loc{loop}(1)-1);
    % offsets for next scan
    xoffset = X_stitch(edge_loc_stitch(end-1));

    % For Display: Plot Trimmed Scan 1
    figure
    plot(X_stitch,Y_stitch,'.')
    axis([0,5000,0,180])
    %ylim([0,180])
    %title('Oil Shim 3: Example Scan 1 Trimmed')
    xlabel('x (\mum)'), ylabel('y (\mum)')
    grid on

else
    %% Determine points to take from current scan
    % Cut off the ends of the scan
    xtemp = x(edge_loc{loop}(1):edge_loc{loop}(end));
    ytemp = y_levelled(edge_loc{loop}(1):edge_loc{loop}(end));
    % Adjust the edge_loc vector to reflect the cutting
    edge_loc_temp = edge_loc{loop}-(edge_loc{loop}(1)-1);

    % For Display: Plot next scan
    figure
    plot(xtemp,ytemp,'.r')
    axis([0,5000,0,180])
    %ylim([0,180])
    %title('Oil Shim 3: Example Scan 2 Trimmed')
    xlabel('x (\mum)'), ylabel('y (\mum)')
    grid on

    % For Display: Plot next scan with previous scan
    % figure
    % plot(X_stitch,Y_stitch,'.'), hold on
    % plot(xtemp,ytemp,'.r')
    % %title('Oil Shim 3: Example Scans 1 and 2')
    % xlabel('x (\mum)'), ylabel('y (\mum)')
    % grid on
    % legend('Scan 1', 'Scan 2')

    % Shift the x values so the first starts at x = 0
    xtemp = xtemp-min(xtemp);
    % Apply x offset from previous scan
    xtemp = xtemp+xoffset;
    % Reset x offset

```

```

xoffset = xtemp(edge_loc_temp(end-1));

% For Display: Plot x shift
figure
plot(X_stitch,Y_stitch,'. '), hold on
plot(xtemp,ytemp,'r.')
xlabel('x (\mum)'), ylabel('y (\mum)')
grid on
%title('Oil Shim 3: x-Shifted Scan 2')

% For Display: Plot y levels and shifts
figure
plot(X_stitch,Y_stitch,'. '), hold on
plot(xtemp,ytemp,'r.')
plot([min(X_stitch),max(X_stitch)],ylevel(loop-
1)*[1,1], 'm')
plot([min(xtemp),max(xtemp)],ylevel(loop)*[1,1], 'g')
xlabel('x (\mum)'), ylabel('y (\mum)')
grid on
%title('Scan Levels')

%% Level current scan with the first scan
% If a scan needs to be moved up or down, the signs will take
care
% of it. Neat!
ytemp = ytemp + (ylevel(1)-ylevel(loop));

% For Display: Scan 2 leveled with Scan 1
figure
plot(X_stitch,Y_stitch,'. '), hold on
plot(xtemp,ytemp,'.r')
%plot([min(X_stitch),max(xtemp)],ylevel(loop-1)*[1,1], 'm')
xlabel('x (\mum)'), ylabel('y (\mum)')
grid on
%title('Scan 2 Leveled with Scan 1')

%% Merge points
% It is unlikely that the two merging sets will
% have the same number of points. Therefore, the smaller
number of
% points is taken for each merging side.
% The average of the two x and y sets will be used for the
stitch.
grab_n = min([right_length(loop-1),left_length(loop)]);

% Previous scan, right side
if right_length(loop-1) > grab_n
    offset = right_length(loop-1)-grab_n;
    xr = X_stitch(end-(right_length(loop-1)-1):end-offset);
    yr = Y_stitch(end-(right_length(loop-1)-1):end-offset);

```



```

else
    xr = X_stitch(end-(grab_n-1):end);
    yr = Y_stitch(end-(grab_n-1):end);
end
% Current scan, left side
xl = xtemp(1:grab_n);
yl = ytemp(1:grab_n);

% Determine the means in x and y
% Most vectors in this program are column vectors
x_merge = mean([xr,xl],2);
y_merge = mean([yr,yl],2);

%% Test Code - Intermediate stitching output code
disp(['Right Length: ',num2str(right_length(loop-1))])
disp(['Left Length: ',num2str(left_length(loop))])
disp(['Grab: ',num2str(grab_n)])
% disp(['# Right: ',num2str(length(xr))])
% disp(['# Left: ',num2str(length(xl))])
% disp('Merge fits within two scans:')
% disp(['Left: ', num2str(x_merge(1) >
X_stitch(edge_loc_stitch(end-1)-1))])
% disp(['Right: ', num2str(x_merge(end) <
xtemp(edge_loc_temp(2)+1))])

%% Test Code - Visualize intermediate stitching steps
figure
% Previous scan
plot(X_stitch,Y_stitch,'. '), hold on
% Current scan
plot(xtemp,ytemp,'.r')
axis([4360,4760,167.4,169.4])

% % Test for proper indexing
%
%plot(xtemp(edge_loc_temp(1):edge_loc_temp(2)),ytemp(edge_loc_temp(1)
:edge_loc_temp(2)),'.g')
% % Level line
% plot([0,max(xtemp)],ylevel(1)*[1,1],'m')

% Edge locations
% plot(X_stitch(edge_loc_stitch(end-
1:end)),Y_stitch(edge_loc_stitch(end-1:end)),'o')
%
plot(xtemp(edge_loc_temp(1:2)),ytemp(edge_loc_temp(1:2)),'or')
% % Top points taken from both sides
% plot(xr,yr,'b')
% plot(xl,yl,'r')
% % Merge results
% plot(x_merge,y_merge,'.m')

```

```

% Axes labels and misc.
grid on
xlabel('x (\mum)'), ylabel('y (\mum)')
set(gcf, 'OuterPosition', [151,282,950,506])

%% Test Code - Visualize final stitching
figure
%plot(X_stitch,Y_stitch,'.')
plot(X_stitch(1:end-right_length(loop-1)),Y_stitch(1:end-
right_length(loop-1)),'.b'), hold on
plot(x_merge,y_merge,'.m')
plot(xtemp(grab_n+1:end),ytemp(grab_n+1:end),'.r')
grid on
set(gcf, 'OuterPosition', [151,282,950,506])
xlabel('x (\mum)'), ylabel('y (\mum)')
%axis([X_stitch(end-right_length(loop-1))-10,X_stitch(end-
right_length(loop-1))+90,...
%      Y_stitch(end-right_length(loop-1))*0.92,Y_stitch(end-
right_length(loop-1))*1.03])

%% Update Stitch
X_stitch = [X_stitch(1:end-right_length(loop-1)); x_merge;
xtemp(grab_n+1:end)];
Y_stitch = [Y_stitch(1:end-right_length(loop-1)); y_merge;
ytemp(grab_n+1:end)];
% update edge_loc_stitch
edge_loc_stitch = [edge_loc_stitch(1:end-
1);edge_loc_temp(2:end)+edge_loc_stitch(end-1)-1];
% Plot the edges to verify
%
plot(X_stitch(edge_loc_stitch),Y_stitch(edge_loc_stitch),'*g')
end

%% Save Figures
%figurecount = length(findobj('Type','figure'));
% figurename = {'Surface Separation';'Channel Line';'dx of
Tops'};
%
% for i = 1:3
%     figure(i)
%     savename = [shim_name,' ',num2str(loop),' ',figurename{i}];
%     %if i ~= 4
%         set(i,'OuterPosition',[85, 323, 1116, 470]);
%     %end
%     print(i,[savedir,savename],'-djpeg','-r300')
% end
% close all

end
toc

```

```

%% Save Stitch
% Reassign variable names for simplicity
% x = X_stitch; y = Y_stitch;
% units = 'um';
% save([filedir,shim_name,' ',num2str(num_scans),' Scan
Stitch.mat'],'x','y','units')
% disp('Stitch saved.')

%% Figure Resizing for Presentations (Using Examples)
% for i = 1:9
% %     if i <= 7 %&& i ~= 3
% %         set(get(i,'CurrentAxes'),'XLim', [0,5200],'YLim',[0,180])
% %     end
% %     set(i,'OuterPosition',[151,282,950,506])
% end
% % Use set(gcf,'OuterPosition',[151,282,950/2,506]) for example
figures in
% % papers

%% fin!
disp('Finished!')

```

C.3 Velocity Interpolation

C.3.1 Velocity_Profile_Interpolator.m

```
% Velocity profile interpolator
% This program takes the six line scans of velocity and interpolates
to
% create a full profile.
% Since the horizontal and vertical scans do not match at the points
they
% should (likely due to a measurement resolution issue), the vertical
scans
% are shifted to equal the points of the horizontal scans.
% James Yih
% 8/18/11
clear, clc, close all
filedir = 'C:\Users\James\Working Home\Graduate Work\Heat Recovery
Unit\Data Processing\Hot Wire\Channel Velocities\';
filename = {'Horizontal Bottom'; 'Horizontal Center'; 'Horizontal
Top';
           'Vertical Left'; 'Vertical Center'; 'Vertical Right'};
% Only 45 channels are loaded, instead of 46 for the vertical scans
since
% the last point seems to be near zero.
% The initialization of VH, hrows, and V are altered to reflect this.
% Also, when the vertical scans are assigned to the matrix V, the
last
% point is omitted.
VH = NaN(45,23); % channel matrix
hrows = [1,23,45]; % rows to place the horizontal scans
vcols = [1,12,23]; % columns to place vertical scans
H = zeros(3,23); % the matrix that contains the horizontal scan
data
V = zeros(45,3); % the matrix that contains the vertical scan data
H_u = zeros(3,23); % uncertainties in horizontal scans
V_u = zeros(45,3); % uncertainties in vertical scans
Vs = V; % the matrix containing the shifted vertical
scans
% Key:
% H(1,:) = Horizontal Bottom      V(:,1) = Vertical Left
% H(2,:) = Horizontal Center      V(:,2) = Vertical Center
% H(3,:) = Horizontal Top         V(:,3) = Vertical Right
%% Load the files
for i = 1:size(filename,1)
    load([filedir,filename{i},'.mat']);
    if i < 4
        % Extract horizontal scan data
        H(i,:) = v;
        H_u(i,:) = u_v;
        % Plot the horizontal data
```

```

figure
errorbar(H(i,:),H_u(i,:),'.');
grid on
xlabel('Channel Number'), ylabel('Maximum Velocity (m/s)')
axis([0,24,-2,26])
set(gcf,'Name',filename{i})
set(gca,'XTick',[0:2:24],'YTick',[-2:2:26])
title(filename{i})
else
    % Extract vertical scan data
    V(:,i-3) = v;
    V_u(:,i-3) = u_v;
end
end

%% Build the data matrix
p1 = zeros(3,2); p2 = p1; % initialize line coefficients matrices
x1 = hrows(1):hrows(2); % x coordinate for line 1
x2 = hrows(2):hrows(3); % x coordinate for line 2
L1 = zeros(length(x1),1);
L2 = zeros(length(x2),1);
for i = 1:size(filename,1)
    %load([filedir,filename{i},'.mat'])
    if i < 4
        % Load horizontal lines
        VH(hrows(i),:) = H(i,:);
    else
        % Determine the how far to shift the vertical scans
        a = H(1,vcols(i-3))-V(hrows(1),i-3);
        b = H(2,vcols(i-3))-V(hrows(2),i-3);
        c = H(3,vcols(i-3))-V(hrows(3),i-3);
        % Line 1 - Bottom and Center Horizontal Scan points
        p1(i-3,:) = polyfit(hrows(1:2),[a,b],1);
        % Line 2 - Center and Top Horizontal Scan points
        p2(i-3,:) = polyfit(hrows(2:3),[b,c],1);

        % Evaluate the lines
        L1(:,i-3) = polyval(p1(i-3,:),x1);
        L2(:,i-3) = polyval(p2(i-3,:),x2);

        % Shift the vertical scan
        Vs(hrows(1):hrows(2),i-3) = V(hrows(1):hrows(2),i-3)+L1(:,i-3);
        Vs(hrows(2)+1:hrows(3),i-3) = V(hrows(2)+1:hrows(3),i-3)+L2(2:end,i-3);

        VH(:,vcols(i-3)) = Vs(:,i-3);
    end
end
end

```

```

%% Plot the vertical shifts
% for i = 1:3
%     figure
%     plot(V(:,i),'.')
%     hold on
%     plot(Vs(:,i),'r*')
%     grid on
%     xlabel('Channel Number'), ylabel('Velocity (m/s)')
%     set(gcf,'Name',filename{i+3})
%     legend('Before Shift','After Shift','location','SouthEast')
%     ylim([10,22])
% end

%% Plot the shift lines
% figure
% cspec = {'-.r','.-g','--'};
% for i = 1:3
%
plot([x1,x2],[polyval(p1(i,:),x1),polyval(p2(i,:),x2)],cspec{i}),
hold on
% end
% grid on
% xlabel('Channel Number'), ylabel('Velocity (m/s)')
% legend('Vertical Left', 'Vertical Center', 'Vertical
Right','location','best')

%% Plot shift lines and data before and after shift
for i = 1:3
    figure
    errorbar(V(:,i),V_u(:,i),'.')
    hold on
    errorbar(Vs(:,i),V_u(:,i),'rx')
    plot([x1,x2],[polyval(p1(i,:),x1),polyval(p2(i,:),x2)],'-')
    grid on
    xlabel('Channel Number'), ylabel('Velocity (m/s)')
    legend('Before Shift', 'After Shift', 'Shift
Line','location','South')
    axis([0,45,-2,26])
    set(gca,'YTick',[-2:2:26])
    set(gcf,'Name',filename{3+i})
    title(filename{3+i})
end

%% Interpolate and Plot
VH = inpaint_nans(VH,3);
[X,Y] = meshgrid([vcols(1):vcols(3)],[hrows(1):hrows(3)]);
% figure
% mesh(X,Y,VH)
% xlabel('X Channels'), ylabel('Y Channels'), zlabel('Velocity
(m/s)')

```

```

%% % title('Method 3')

VHa = Average_Matrix(VH,1);
figure
mesh(X,Y,VHa)
xlabel('X Channels'), ylabel('Y Channels'), zlabel('Velocity (m/s)')
zlim([0,25])

figure
contourf(X,Y,VHa)
colorbar
xlabel('X Channels'), ylabel('Y Channels')
title('Velocity (m/s)')
grid on

%% Calculate the volume flow rate
A = 1.6*.89/1000^2;      % channel area
vol = sum(A*VH(:))      % volume flow rate, m^3/s
vol_cfm = vol*2119      % volume flow rate, cfm

% Fin!

```

C.3.2 inpaint_nans.m

inpaint_nans.m was used in the velocity interpolation to fill in the NaN elements of the 2-D array. The function file was written by John D'Errico and can be found at <http://www.mathworks.com/matlabcentral/fileexchange/4551>. Sparse linear algebra and PDE discretizations were used to determine the solution.

C.4 Common Function Files

C.4.1 Central_Difference.m

```
function dy = Central_Difference(x,y)
% This function uses central differencing to compute the first
% derivative dy/dx. The end points are handled with forward and
backward
% differencing schemes.
% x and y are vectors of the same size. x does not need to have a
uniform
% step size.
n = length(x);
dy = zeros(size(x));
for i = 1:n
    if i == 1
        dy(i) = (y(i+1)-y(i))/(x(i+1)-x(i));
    elseif i == n
        dy(i) = (y(i)-y(i-1))/(x(i)-x(i-1));
    else
        dy(i) = (y(i+1)-y(i-1))/(x(i+1)-x(i-1));
    end
end

% Fin!
```


C.4.2 Average_Matrix.m

This file was used for smoothing data.

```
function Matrix_Out = Average_Matrix(Matrix_In,Window_Size)
aw = Window_Size; % Averaging window
Matrix_Out = zeros(size(Matrix_In));
[num_row,num_col] = size(Matrix_In);
for row = 1:num_row
    for col = 1:num_col
        if row-aw < 1
            a_1 = 1;          a_2 = row+aw;
        elseif row+aw > num_row
            a_1 = row-aw;    a_2 = num_row;
        else
            a_1 = row-aw;    a_2 = row+aw;
        end

        if col-aw < 1
            b_1 = 1;          b_2 = col+aw;
        elseif col+aw > num_col
            b_1 = col-aw;    b_2 = num_col;
        else
            b_1 = col-aw;    b_2 = col+aw;
        end
        Matrix_Out(row,col) = mean(mean(Matrix_In(a_1:a_2,b_1:b_2)));
    end
end
```

