

University of  
St Andrews

CS4099 MAJOR SOFTWARE PROJECT

---

## Graph matching with Lobsters

---

APRIL 9, 2018

*Supervisor:*  
Kasim Terzić

*Submitted By:*  
Sizhe Yuen

# Abstract

The ability to make measurements on lobsters is important in monitoring the size and health of the creatures. This project explores the representation of lobsters as attributed graphs to discover and measure properties such as their size, using an existing dataset of lobster images. Computer vision techniques are applied to detect interest points of different lobster body parts. Probabilistic models and subgraph matching are then used to label and build up the graph representation. The advantages and disadvantages of the various methods and algorithms used are discussed before the final methods were chosen. The proposed approach can be used to replace existing manual methods of measurement to improve the efficiency of both fishers and scientists in the field. The effectiveness of the methods are evaluated against the existing dataset to discover that the use of graph matching is a suitable technique for lobster categorisation. Positive results were acquired for aspects of detection and identification, which shows the viability of the graph matching approach.

# Declaration

I declare that the material submitted for assessment is my own work except where credit is explicitly given to others by citation or acknowledgement. This work was performed during the current academic year except where otherwise stated.

The main text of this project report is NN,NNN\* words long, including project specification and plan.

In submitting this project report to the University of St Andrews, I give permission for it to be made available for use in accordance with the regulations of the University Library. I also give permission for the report to be made available on the Web, for this work to be used in research within the University of St Andrews, and for any software to be released on an open source basis. I retain the copyright in this work, and ownership of any resulting intellectual property.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Objectives . . . . .	1
<b>2</b>	<b>Context Survey</b>	<b>3</b>
2.1	Previous work . . . . .	3
2.2	Related work . . . . .	3
2.3	Graph matching problem . . . . .	4
<b>3</b>	<b>Software Engineering Process</b>	<b>6</b>
3.1	Existing software . . . . .	6
3.2	Technologies used . . . . .	7
3.3	Process . . . . .	8
<b>4</b>	<b>Methodology</b>	<b>9</b>
4.1	Overview . . . . .	9
4.2	Annotation of dataset . . . . .	9
4.3	Keypoint detection . . . . .	11
4.4	Keypoint filtering . . . . .	15
4.5	Subgraph creation . . . . .	19
4.6	Subgraph matching . . . . .	21
4.7	Graph building . . . . .	21
<b>5</b>	<b>Results and analysis</b>	<b>24</b>
5.1	Experimental design . . . . .	24
5.2	Brief qualitative analysis . . . . .	24
5.3	Precision and recall . . . . .	25
5.4	F1 score . . . . .	30
5.5	Classification . . . . .	35
<b>6</b>	<b>Evaluation</b>	<b>40</b>
6.1	Related work . . . . .	40
6.2	Objectives . . . . .	40
<b>7</b>	<b>Conclusion</b>	<b>42</b>
7.1	Discussion . . . . .	42
7.2	Future work . . . . .	42
7.3	Summary . . . . .	43

# 1 Introduction

The fishing industry plays a large role in the Scottish economy. Many rules and regulations have been put in place to improve the management of inshore fisheries in Scotland for sustainable fishing. The minimum landing size is an example of a simple regulation to protect breeding stock, not allowing any catches below the minimum size to be landed. This regulation ensures a proportion of the animal population continue to grow and reproduce [10] and is used particularly for inshore shellfisheries. Lobsters are an example of a family of crustaceans affected by the application of this regulation.

Typically, catches have to be sorted and measured manually in order to adhere to regulations on minimum landing size. Illegal and unwanted products are then discarded back to the sea. Rochet [24] notes that discarding continues to be an important problem in world fisheries as discarded products often do not survive. This defeats the purpose of the regulation to preserve breeding stock and further hurts the industry as discarded products are lost without economic gain.

There has been some research in the area of applying computer vision techniques to automatically identify and detect species of fish [6, 8, 20] and research into automatic measurement of fish size [11, 25] which can be applied to reduce discarding. The detection of fish sizes is the first step in mitigation programs [24] to reduce discards, as Rochet states that a knowledge base of discarding patterns provides a strong basis for discussion on mitigation measures. Much work is focused on fish due to the impractical nature of manual observation and measurement. However, little work has been done that focuses on using such techniques and programmes for lobsters.

In this project, the application of subgraph matching and interest point detection is explored to automatically identify different parts of a lobster such as the claw, body or tail and from such identification, build a complete graph representation of a lobster. The proposed method uses an attributed graph, which represents the body parts as a set of nodes connected by weighted edges to represent the distance between the parts. This was done using a combination of computer vision techniques and probability models. The computer vision techniques identified relevant keypoints that the probabilistic models can label to build into subgraphs. The subgraphs are then matched and built into the complete graph with the same probability basis. Finally, analysis and evaluation was done to measure the performance and accuracy of the proposed methods. Using an existing dataset of lobster images gathered in collaboration with the Scottish Oceans Institute, this project was able to test and evaluate its methods against current attempts to determine the category (mature or juvenile) of lobsters.

## 1.1 Objectives

Primary and secondary objectives of the project are listed below.

### 1.1.1 Primary objectives

- Explore and create suitable graph representations for lobsters
- Match lobster subgraphs with complete graphs using existing software
- Automatically detect interest points from images
- Evaluate this method of graph matching on lobsters against the existing dataset

The objectives are vastly unchanged from the beginning of the project, but the objective *match lobster subgraphs with complete graphs using existing software* has been changed from the original to better reflect what was done. Originally, the objective was to *measure similarity of lobster graphs with existing software*. During the development of the project, this objective moved from the direction of using graph similarity to

using subgraph isomorphism as a means of matching. Hence the original objective was changed to match the change in direction.

In this project, all primary objectives were met, resulting in a complete method which is able to take an image of a lobster from the dataset and:

- Detect interest points
- Label interest points as lobster body parts
- Create a graph representation from the detected points and labels
- Match the graph representation to ideal models for evaluation

The accuracy of the method was then analysed and evaluated.

### 1.1.2 Secondary objectives

- Extend existing or come up with new algorithms for creating lobster graphs from images
- Apply this technique on more complex images with noise such as natural lobsters in their environment
- Apply this technique for video instead of images to give properties and information on the lobster in real time.

Certain secondary objectives were touched upon and investigated during the design and implementation of this project, though none were explicitly explored. The ability to deal with noisy backgrounds was both noted and dealt with as an issue in the filtering (section 4.4) and combination (section 4.7) stages respectively. The final method developed is a novel way to create a graph representation of a lobster from an image, combining a number of existing algorithms.

# 2 Results and analysis

The developed models and methods were able to take images from the dataset and to match it with an attributed graph. The graph contains labels for each node that specifies the body part identified and edges that connect each node. Performance of the developed models and methods is then measured to give numeric results. First, a qualitative evaluation on the results of the matched images and graphs was carried out to show the effectiveness of the matching visually. Then a quantitative analysis using different metrics was performed to evaluate how well the methods used in this project is able to perform against previous methods. The quantitative values measured are later compared to Abdallah's results in section 6. It would be expected for the method developed in this project to give better results.

## 2.1 Experimental design

For quantifying the results, precision and recall metrics were initially used. The goal of using precision and recall is to see what kind of trade-off there is between the relevant keypoints detected in all detected keypoints (precision) and the relevant keypoints detected in all relevant keypoints (recall). The F1 score was then computed to get the harmonic mean of precision and recall. This allows the best labelling and colour histogram thresholds to be found. Finally, by applying a combination of probability and graph edit distance on the models, images from the rest of the dataset were classified as either mature or juvenile.

They are also two other parameters that were explored in parallel with the evaluation metrics:

1. As explained in section 4.7, three methods were investigated for rebuilding the matched subgraphs into the final lobster graph. Although the keypoint method was not evaluated due to its shortcomings, the label and graph methods are both used during the evaluation to test which is more effective.
2. In section 4.2, it was described how the additional annotation of attribute graphs to the lobster images were split into mature and juvenile categories based on the original dataset. The probabilistic models are based on the two sets of annotations and their performance on both mature and juvenile lobsters can be compared. It is also interesting to see if any model performs better with a particular method or for a particular label.

To calculate the precision and recall, the ground truth of the correct lobster graph must be known and so only the subset of images with annotated attributed graphs are used in precision-recall experiments.

All measurements needed for evaluation are deterministic, the probabilities and keypoints detected may change based on thresholds, but will not change between runs. As such, only one run of any of the experiments is needed to get the results needed.

## 2.2 Brief qualitative analysis

A brief qualitative evaluation of the output graph drawn on top of the images is explored here to get a visual glimpse of how well the developed algorithm works.



(a) Example of good match



(b) Example of poor match

Figure 1: Example of final graph matches. In the case of the poor match, a juvenile model was used on an image of a mature lobster, leading to incorrect labelling despite reasonable keypoint detection.

It can be seen that when the algorithm works, the graph is matched very well to each body part on the lobster, with the nodes properly connected. Further, the keypoints are all labelled correctly and only one claw keypoint was missing. In the case where the matching was poor, the labels were often incorrect, parts of the graph are missing (no claws) and irrelevant keypoints are detected and labelled. There are three main reasons for this:

- The keypoint was not detected, so it could not have been matched or labelled
- Thresholds on labelling and histograms filtered out relevant keypoints
- Probabilistic model filtered out correct subgraph configurations

Octave filtering is unlikely to remove relevant keypoints because it only removes keypoints that are too small to be suitable. The latter two reasons are more likely than the keypoint not being detected at all because SIFT is able to detect an abundance of keypoints as demonstrated in section 4.3. The other two cases happen more often when the incorrect model is used, for example a juvenile model used on a mature lobster as in figure 19b. The sizes of nodes and lengths of edges are different for the two models and so mislabelling happens more often. Mislabelling can then cause a knock on effect which affects the matching step and thus the final lobster graph.

### 2.3 Precision and recall

The evaluation for precision and recall was split into two evaluations, one for the performance of identifying correct keypoints and the second for the performance of keypoint labelling. The evaluations are split into two parts because defining false positives and false negatives when evaluating both aspects together is a difficult problem, the key issue being an overlap between false positives and false negatives for correctly detected but mislabelled keypoints. For example, if a keypoint has been correctly identified and labelled as a *claw* but is actually an *arm*, then it should be a false positive because the labelling is incorrect. However, it should also be a false negative because correct *arm* labelling is missed. Because of these difficult definitions, it made sense to split the precision and recall evaluation into two separate parts where the false positives and false negatives could be clearly defined.

In order to create a precision-recall curve, the probability threshold for labelling keypoints and the threshold for colour histogram filtering is incrementally changed. This shows how the two thresholds affect the precision and recall of the lobster graphs created. Note that results for very low thresholds were not obtained due to the large time taken because of the combinatorial explosion. It would then be expected that as the thresholds changes, the trade-off between precision and recall also changes, for example a high labelling threshold may

give lower recall as relevant annotations are missed, but high precision as all keypoints identified in the image are relevant keypoints.

### 2.3.1 Keypoint identification

The precision and recall metrics were calculated for keypoint identification to see how many keypoints from the annotated images could be re-identified in the final graph. The labels of each keypoint and the edges between them were not taken into account for these results. The precision and recall for the evaluation is defined as follows:

$$\text{Precision} = \frac{\text{Correctly detected keypoints}}{\text{Correctly detected keypoints} + \text{Incorrectly detected keypoints}} \quad (1)$$

$$\text{Recall} = \frac{\text{Correctly detected keypoints}}{\text{Correctly detected keypoints} + \text{Missed keypoints in annotation}} \quad (2)$$

False positives are incorrectly detected keypoints, as they have been identified, but are not correct compared to the truth. The false negatives are keypoints that are in the annotation, but were not detected for the final graph and hence incorrectly identified as an irrelevant keypoint. Keep in mind that the dataset was categorised as either a juvenile or mature lobster and two probability models were created based on the two categories. Both models and categories are tested for results, as it is expected that a mismatch between the two would result in lower performance.

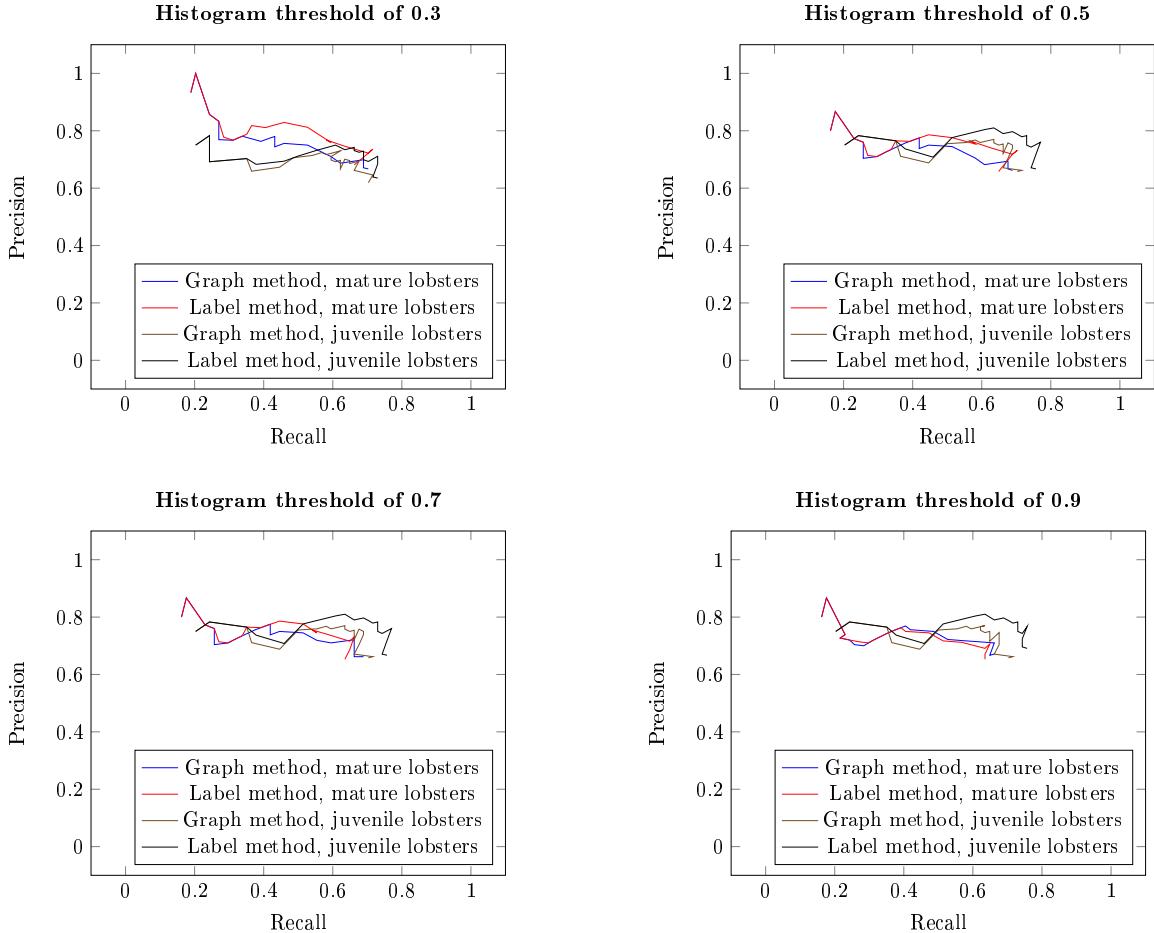


Figure 2: Precision/recall graphs using the mature model with varying label thresholds.

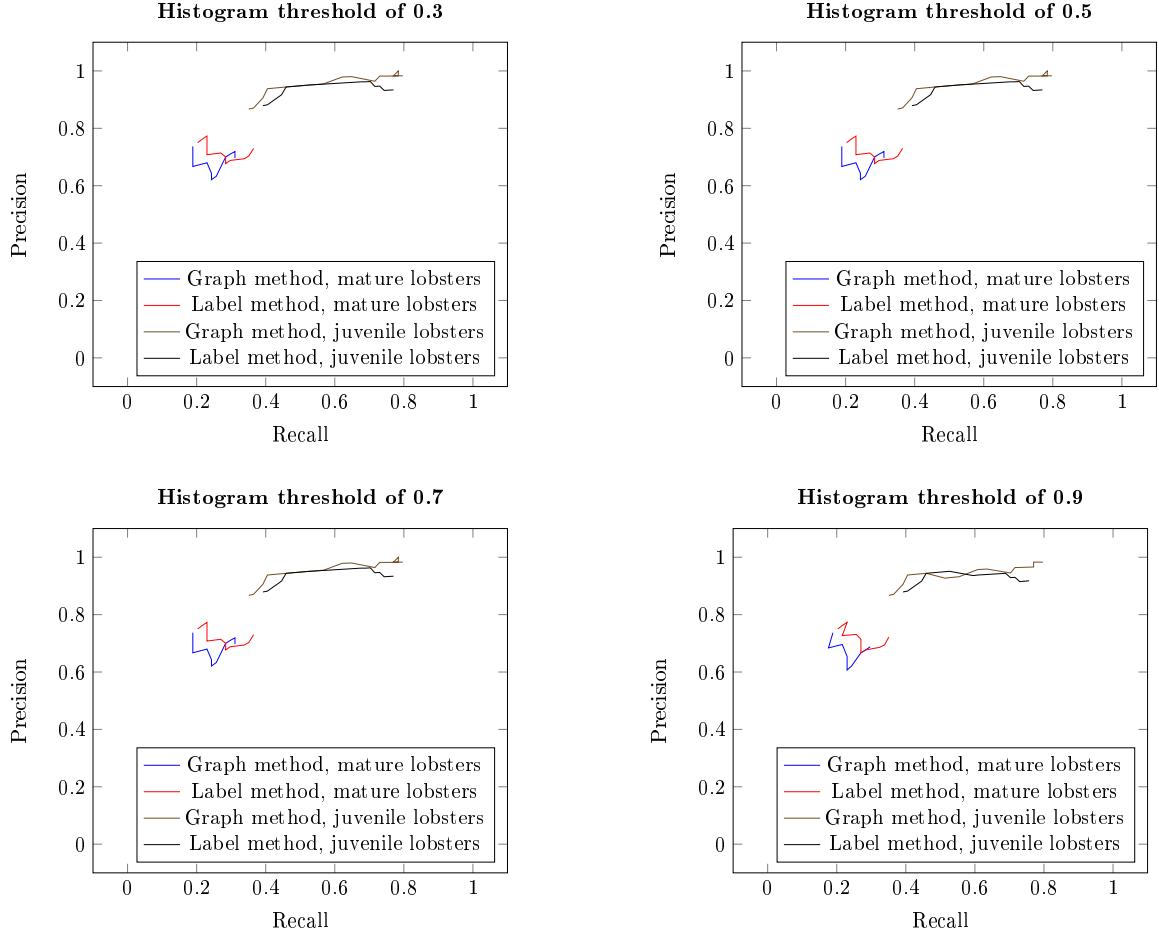


Figure 3: Precision/recall graphs using the juvenile model with varying label thresholds.

It can be seen from figure 21 and 20 that despite the variation of the label and histogram thresholds, there is not a clear precision-recall curve that shows the trade-off between the two metrics. The change in the labelling threshold clearly affects the recall, but precision seems independent of both the recall and thresholds as it stays relatively the same even though recall improves dramatically. This suggests there is either no inherent trade-off between precision and recall in this problem, or the thresholds being changed is not suitable to find the trade-off and therefore does not affect it.

The varying label and histogram filtering thresholds may not have provided useful functions as they only indirectly affect the final graph that is built. The two thresholds significantly change what labels and keypoints are kept in the stages leading up to the creation of the final graph, for example a high label threshold would mean less subgraphs to explore as less labels could be applied. However it is the probabilistic model of choosing subgraphs which ultimately affects the lobster graph that is created. The issue here is that there is no threshold that can be applied to this probabilistic model, as the most probable subgraphs are chosen incrementally. For this reason, examining the precision and recall metrics is not suitable to evaluate performance.

Another interesting thing to note here is the difference between using a mature lobster model (figure 20) and using a juvenile lobster model (figure 21). There is a clear difference in both precision and recall when applying the juvenile model to the two categories of images. However, the difference is less obvious with the mature model. The smaller difference may come from the increased variance in the mature lobster images for sizes of keypoints and length of edges. On the other hand, the juvenile lobster images are more consistent

with the lobster sizes.

### 2.3.2 Keypoint labelling

In the evaluation of keypoint labelling, the precision-recall calculations are altered slightly because only the labels on the detected keypoints matter here, rather than the detection itself. Additionally, the evaluation can be focused on each specific label (claw, tail, etc.) to see if there are any strengths or weaknesses for specific body parts. Precision and recall for here are defined as follows:

$$\text{Precision} = \frac{\text{Correctly labelled keypoints}}{\text{Correctly labelled keypoints} + \text{Incorrectly labelled keypoints}} \quad (3)$$

$$\text{Recall} = \frac{\text{Correctly labelled keypoints}}{\text{Correctly labelled keypoints} + \text{Missed labels in annotation}} \quad (4)$$

Incorrectly labelled keypoints are either ones where the keypoint was wrong, in which case the label is incorrect by default, or the keypoint is correct, but the label is wrong compared to the annotations. These are classified as false positives. The false negatives are labels which were missed in the annotations, for example if there were two claw labels in the annotations but only one claw was found in the final graph.

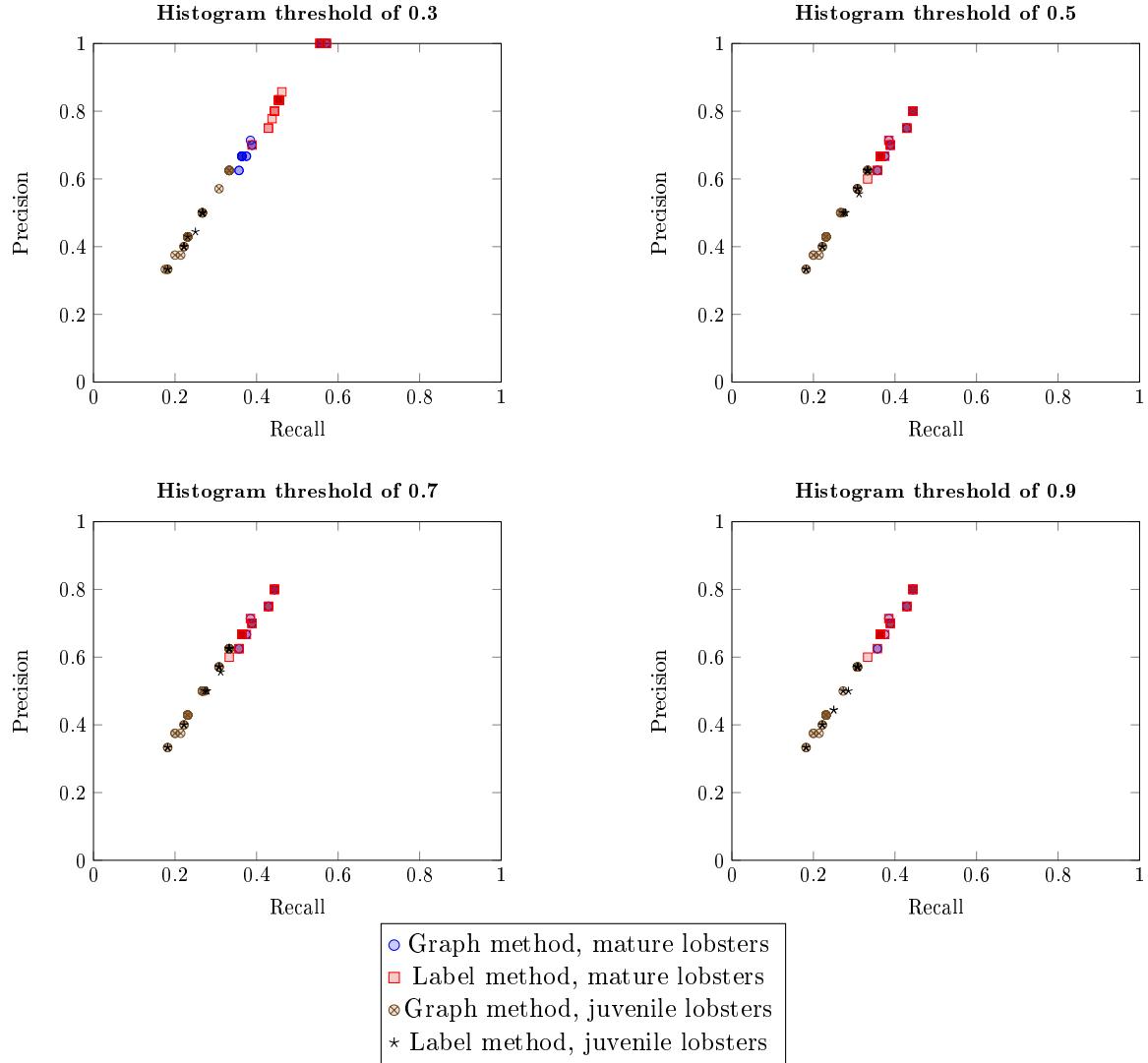


Figure 4: Precision/recall graphs with the mature model for label *claw*.

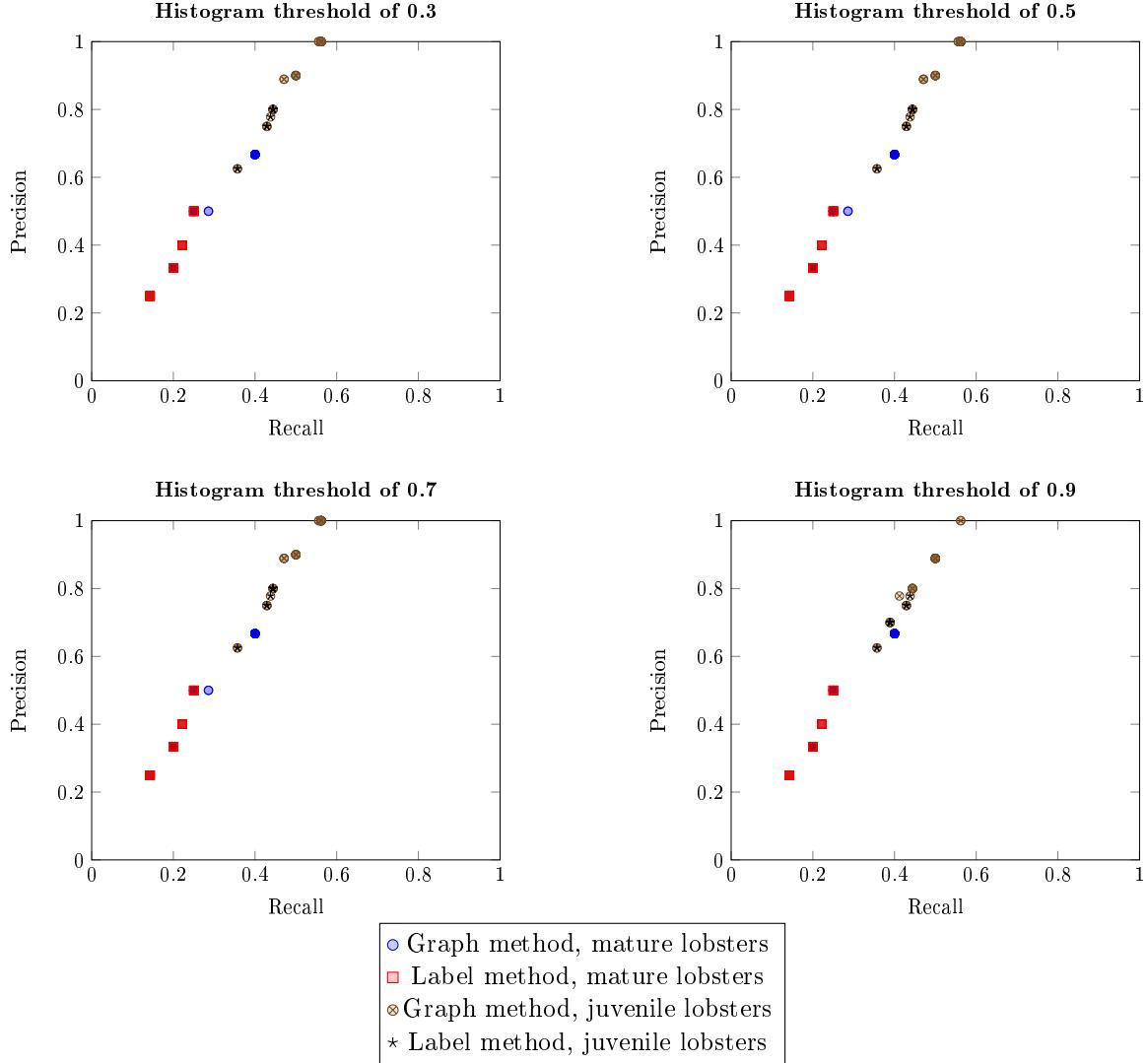


Figure 5: Precision/recall graphs with the juvenile model for label *claw*. Further graphs of all other labels can be found in appendix ??

The precision-recall graphs for labels confirm that precision and recall are not very suitable metrics due to the lack of a good threshold to alter. There seems to be something missing in the curves, as the linear relationship from varying label thresholds and different methods and models is not expected from a precision-recall curve. Potentially each point in the graphs represent a separate precision-recall curve, but the threshold required to be varied that gives the trade-off and the rest of the curve could not be found.

## 2.4 F1 score

Although precision and recall did not give much insight into the models and methods used, the metrics can still be used to empirically find the best labelling and histogram filtering thresholds that should be used for classification. The F1 score can be used as a weighted average of precision and recall where both contribute equally. The scores ranges from 0 to 1, where 1 means perfect precision and recall. The best thresholds are therefore those whose F1 score is the highest. The F1 score also gives a reasonable look into which models and methods are the most effective. The identification and labelling evaluations continue to be kept separate for the same reason as mentioned above.

#### 2.4.1 Keypoint identification

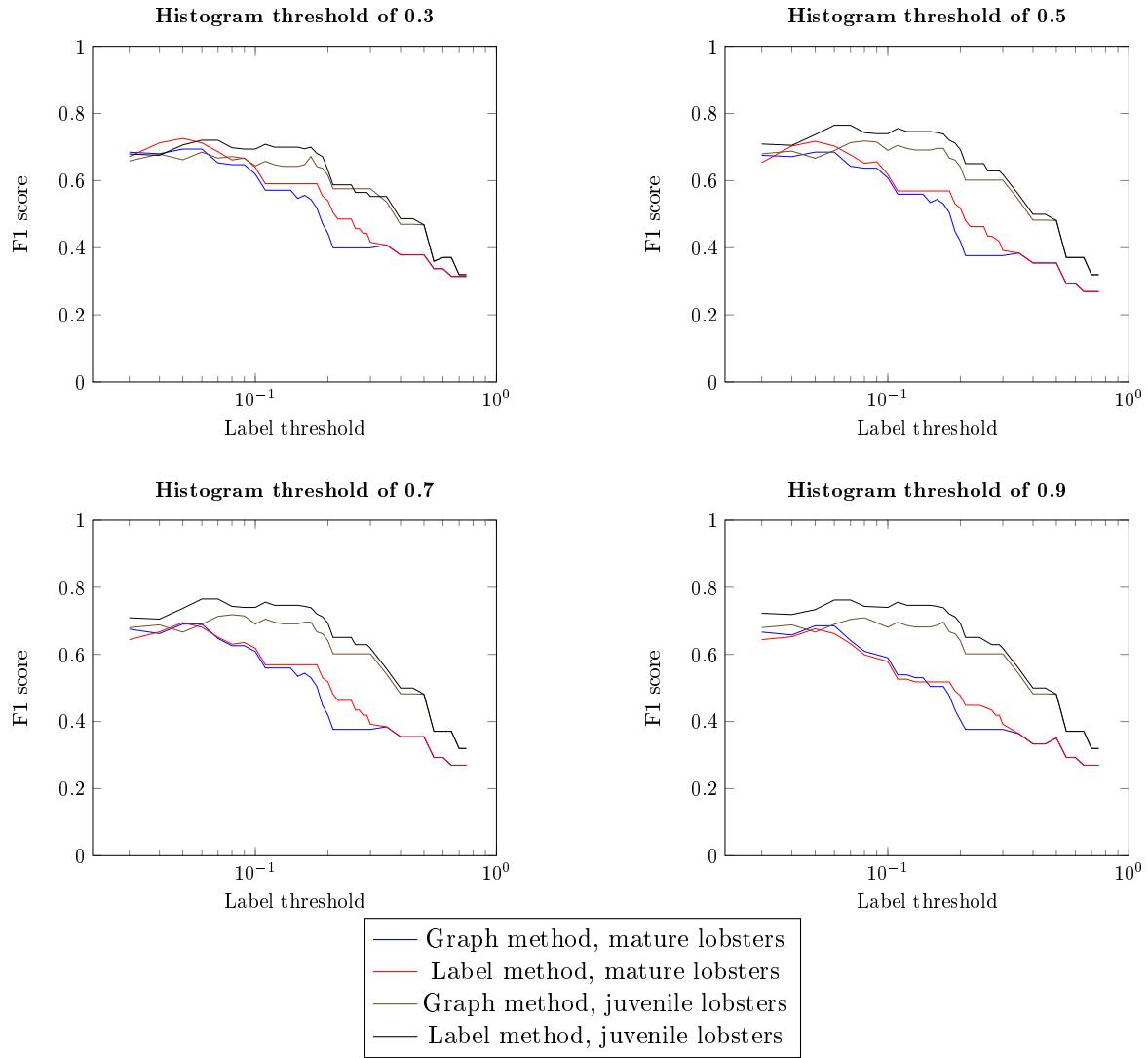


Figure 6: F1 score for varying thresholds with the mature model.

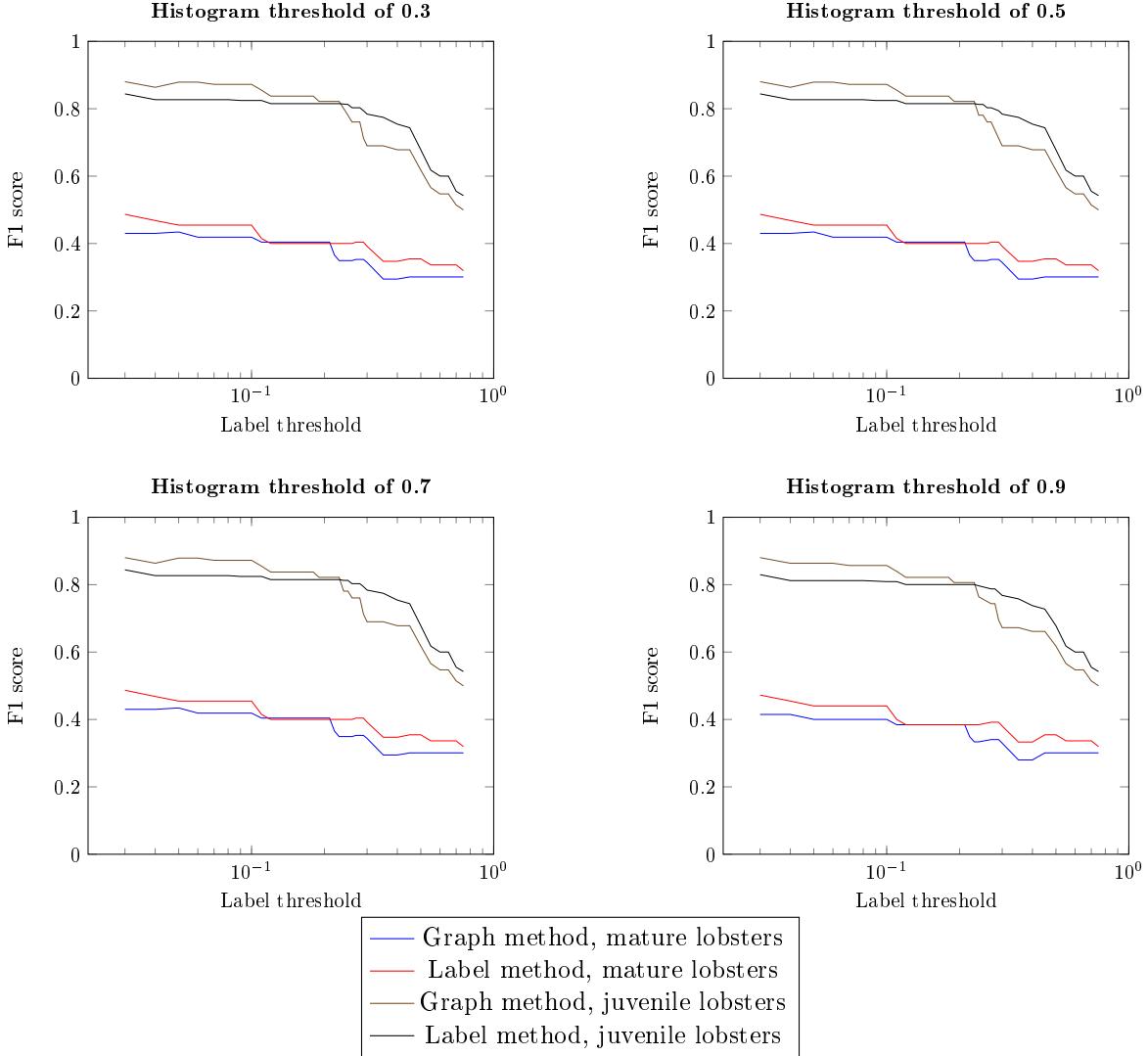


Figure 7: F1 score for varying thresholds with the juvenile model.

Method	Model	Category	Label threshold	Histogram threshold	F1 score
Graph	Juvenile	Juvenile	0.03	0.3-0.9	0.880
Label	Juvenile	Juvenile	0.03	0.3-0.7	0.844
Label	Mature	Juvenile	0.07	0.5-0.7	0.765
Label	Mature	Mature	0.05	0.3	0.726
Graph	Mature	Juvenile	0.08	0.5-0.7	0.718
Graph	Mature	Mature	0.06-0.05	0.3	0.694
Label	Juvenile	Mature	0.03	0.3-0.7	0.487
Graph	Juvenile	Mature	0.05	0.3-0.7	0.434

Table 1: Best F1 score and thresholds for different combinations of methods and models.

In general, it is shown clearly that the label threshold has a substantial impact on performance. As the label threshold increases, the F1 score decreases because it becomes less probable for any particular label to be applied to a keypoint. The decreased number of labelled keypoints then cause less subgraphs to be matched and so reduces the number of final keypoints. Moreover, at high label thresholds, it becomes more likely

that a keypoint will not be labelled at all and so additional relevant information is lost.

The histogram threshold seems to have less of an impact as little difference can be seen between each graph with a different histogram threshold. This is interesting because with a low histogram threshold, more irrelevant keypoints will be kept so it would be expected that the F1 score decreases. The lack of an impact by the histogram threshold has two possible explanations which are not mutually exclusive:

1. The colour histogram filtering is very effective such that irrelevant keypoints require a very low threshold to be included.
2. The probabilistic models are effective in removing irrelevant keypoints, so the resulting lobster graphs are still the same or very similar even when more keypoints are introduced.

Of the two explanations, the latter seems more possible. As demonstrated in section ??, a colour histogram filter can still result in irrelevant keypoints being detected. As such, the lower threshold likely adds more keypoints to be matched, but the matching steps and the building of further probabilistic graphs are robust against such noise, which is why little difference can be seen between the different histogram thresholds.

The graphs of figure 24 and 25 further demonstrate juvenile lobsters being better detected in general, regardless of the model or method used, when compared to mature lobsters. Moreover, a high F1 score greater than 0.8 can be obtained for keypoint identification, showing the strength and success of the approach in combining computer vision techniques with graph matching.

#### 2.4.2 Keypoint labelling

The F1 score is also looked into for the labelling to see if any particular labels perform better. This would be interesting to see if certain labels are detected with higher certainty compared to others, showing the strengths and weaknesses for each method and model.

<b>Label</b>	<b>F1 score</b>
Body	0.99
Tail	0.70
Head	0.58
Arm	0.57
Claw	0.45
Back	0.34

Table 2: Average F1 score for each label, averaged over all over parameters

The average F1 score for all labels can be seen in table 2. The average is used to show which labels can be detected and labelled correctly the most regardless of the methods or thresholds. The body keypoint is found very reliably, almost 100% of the time. Because subgraphs of 3 nodes were chosen as the intermediate between matching and rebuilding the complete graph, finding the body keypoints is very important as every 3 node subgraph must contain a body. In cases where the body is not detected, no final graph can be created as no subgraphs will be matched.

The tail label can also be found reasonably and this can be expected as it is a very distinct feature on the lobster. The ability to discover the tail node is key for tail width measurement, which can be important depending on what features of the lobster need to be measured. Interestingly, the arms and claws are not as easily found despite also being distinct features. This could be due to other irrelevant keypoints found on the lobster that match the same size and distance of the claws and arms to the body, causing incorrect matches.

Label	F1 score	Label threshold	Histogram threshold
Body	1.0	0.3-0.1	0.3-0.9
Tail	0.94	0.35	0.3-0.9
Head	0.75	0.55	0.9
Arm	0.66	0.6	0.3
Back	0.5	0.27-0.29	0.3-0.9
Claw	0.49	0.45	0.3

Table 3: Best F1 score for each label, averaged over all methods, models and categories to show the best thresholds that can be used for each label.

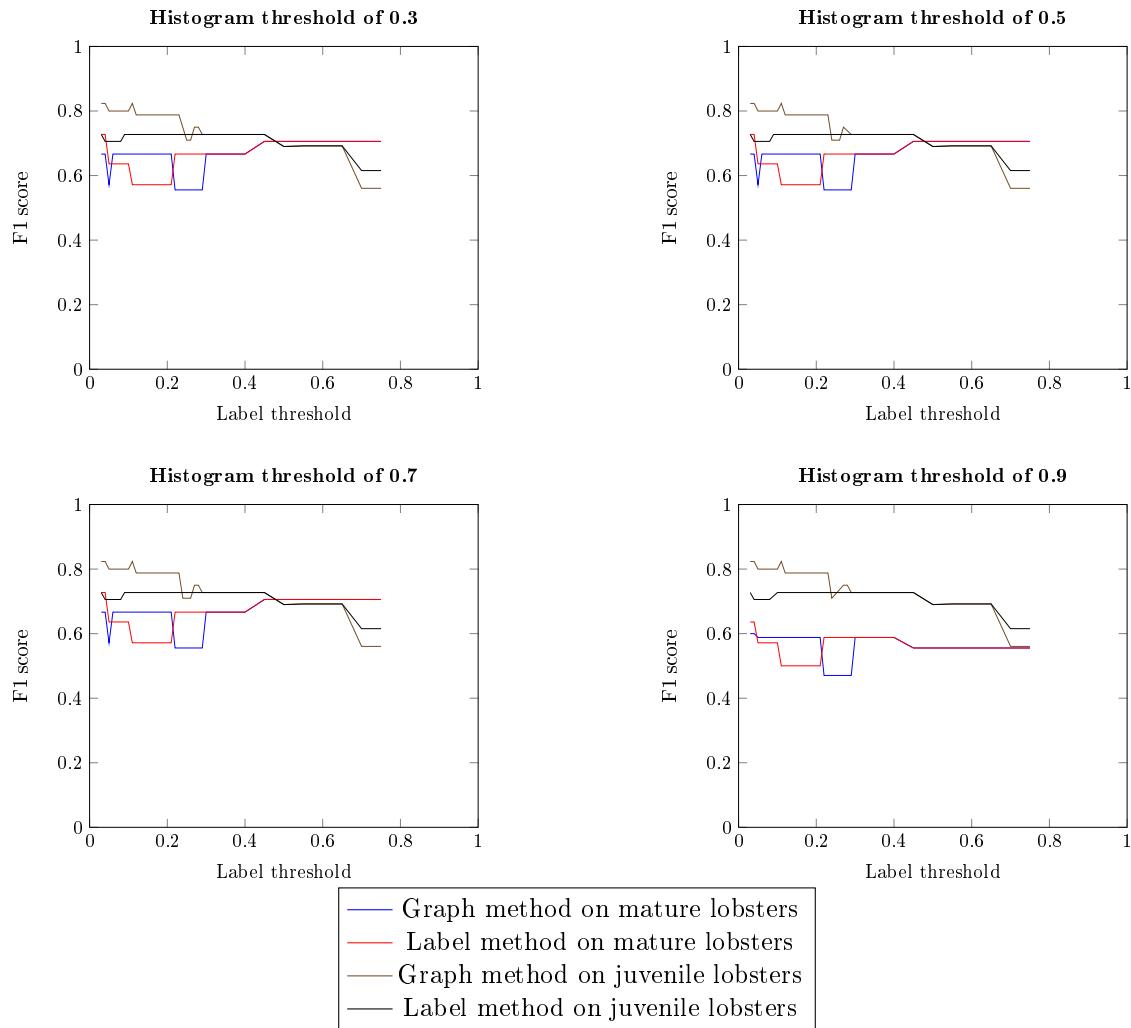


Figure 8: Juvenile lobster model on the arm label

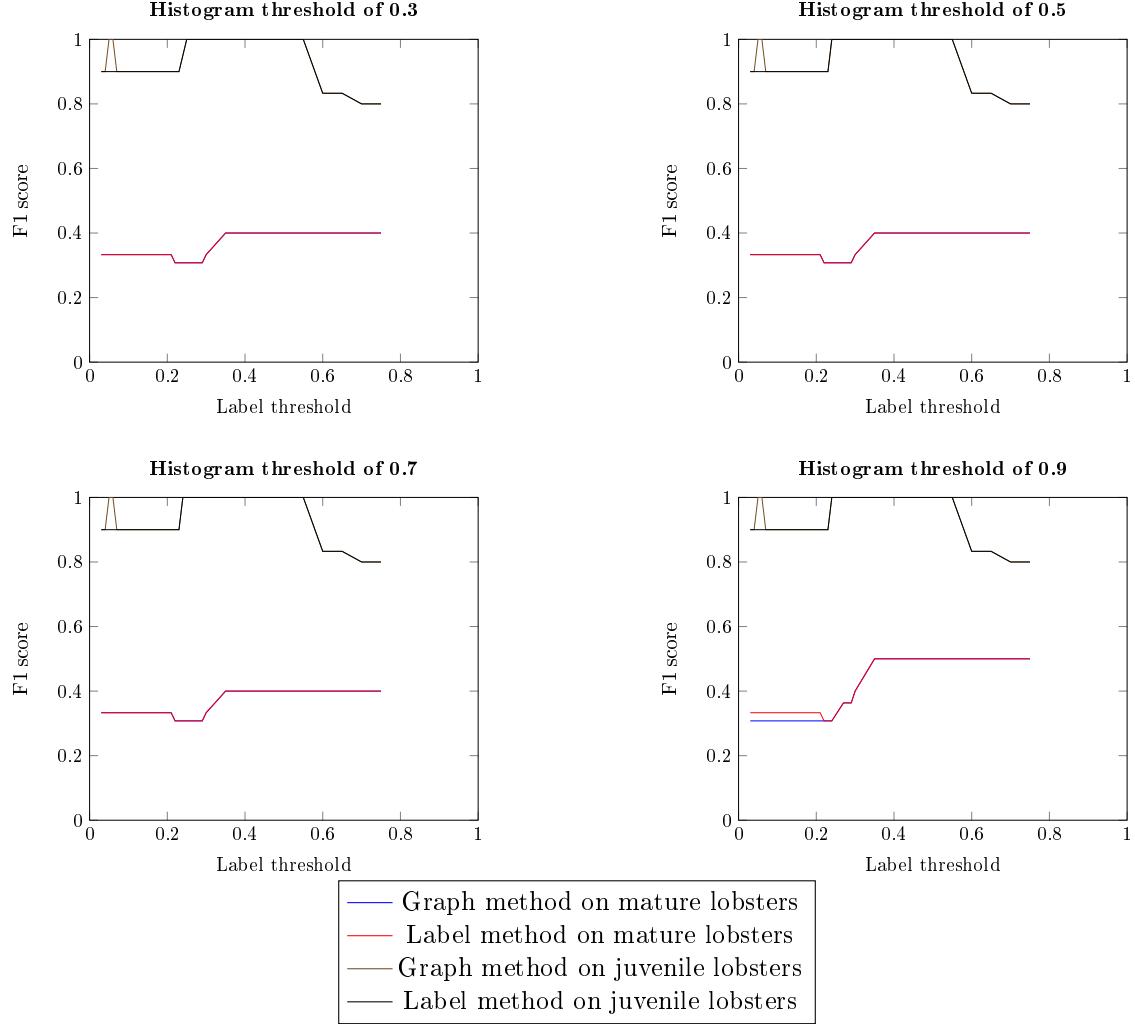


Figure 9: Juvenile model, head label

The range of thresholds shown in table 3 and the varying performance of different methods shown in figures 26 and 27 indicates that different labels are affected by different thresholds and methods. For example, the model used for the arm label has little effect on its F1 score as there is lots of overlap between the mature and juvenile models. However, the model used for the head label has a much larger effect, where the correct model (juvenile model on juvenile lobster) massively outperforms itself against incorrect lobster categories. Further, unlike the F1 scores for keypoint identification, it is not clear that lowering the label threshold lowers the score. The complicated issue with the combination of all the different parameters is interesting for further exploration, especially for labels with lower average F1 score such as the claw or arm.

## 2.5 Classification

As both a mature and juvenile model are developed to match to the lobster images, how well each model is matched to the lobster in the image can be used for classification. For example, if the image contains a mature lobster, then the mature model would be expected to be better matched compared to the juvenile model.

As a probabilistic model was developed and used for the creation and matching steps to produce the final

matched lobster graph, the use of probabilities is again explored for classification. To use probabilities for classification, the product of the probabilities for each node and edge can be calculated as was done in section 4.7. This would give two probabilities, one for each graph created by the mature and juvenile models. However, in cases where nodes and edges were not identified, the probability of the graph would increase because the probabilities are multiplied. This is the opposite of what is desired, as missing nodes and edges should incur heavy penalties. Furthermore, there is no easy way give a penalty to missing nodes and edges without being too arbitrary. Therefore, using just the probabilities is not a very suitable way to define the two classifications.

With the two final graphs of mature and juvenile lobsters, a graph similarity score can be computed. The graph edit distance is typically used as a distance measure between two graphs. It is defined as the weighted sum of the cost of edits such as node insertion or edge deletion required to convert one graph to the other. The maximum common subgraph as described in section 2.3 is another concept that can be applied. The larger the maximum common subgraph between the matched graph and ideal graph, the better the match. There are a variety of such algorithms for graph similarity and matching that could be further explored [17]. In [12], Bunke proved that by using a particular cost function, the computation for the graph edit distance is equivalent to the maximum common subgraph problem. The problem with using edit distance metrics is that it does not take into account errors in labelling and matching. In other words, a matched lobster graph can have a very small distance metric to the ideal lobster graph because most of the body parts and edges were identified, however the body parts could be mislabelled and the edit distance would not be able to allow for this issue.

As such, a combination of the probability and graph edit distance must be used to determine how well a detected graph matches to an ideal graph to classify images without ground truths. To take the graph edit distance into account with the probabilities, the final probability of the graph is divided by edit distance. Missed matches will increase the edit distance, so the increased probability due to missed matches is balanced with the additional edit distance. A constant multiplier is used to help with the balance of the equation, which was found experimentally.

### 2.5.1 Validation

The classification is first tested on the annotated dataset to see how well it can perform. For unclassified cases, the lobster graph could not be created at all and so the image could not be classified.

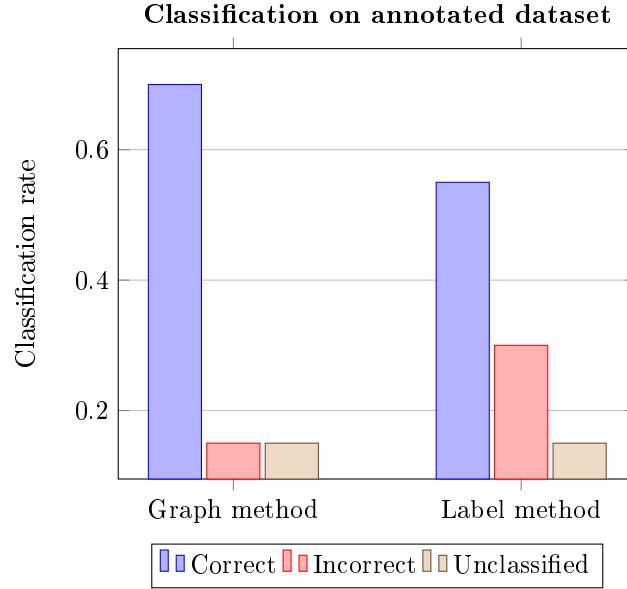


Figure 10: Overall classification rate on annotated images

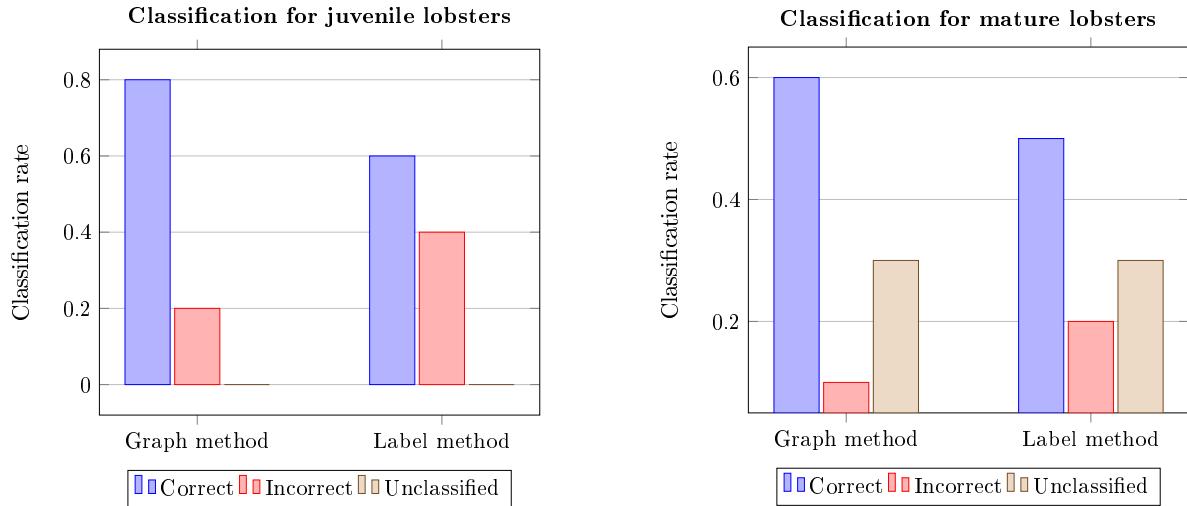


Figure 11: More detailed classification results on annotated images on the specific lobster categories.

It can be seen that the graph method slightly outperforms the label method and gets few images incorrect. This makes sense as the graph method takes into account the node connections whereas the label method is only concerned with individual keypoint labels. Further, these results continue to support the finding that the juvenile model works better than the mature model. This is especially due to the images that are not classified in the mature model. This shows that the methods used can be quite accurate as long as the probability distributions in the models are consistent and the proper keypoints can be found and identified. The case for unclassified images comes the lack of graph that could be built, likely due to the lack of labelled keypoints. The weakness in the methods therefore comes from the detection and labelling, rather than the final graph building, as the graph method does not have a high misclassification rate.

### 2.5.2 Testing

The same experiments are repeated for the rest of the dataset which has not been used up until this point. Note that there is a bias towards mature lobsters (67%) compared to juvenile lobsters (33%) in the full dataset.

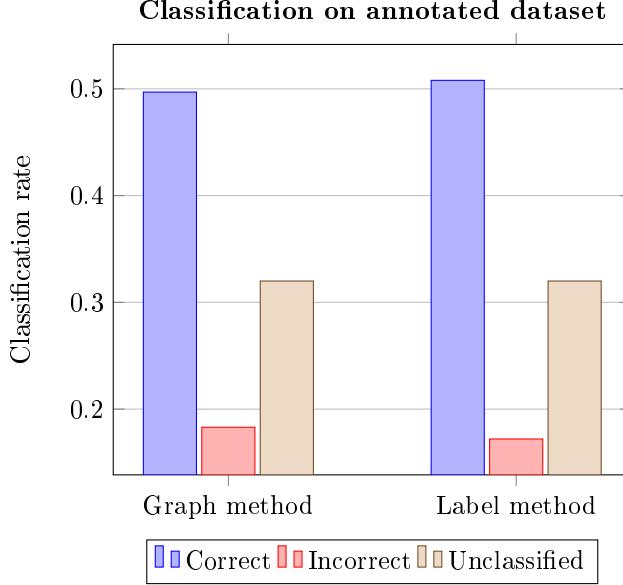


Figure 12: Overall classification rate

The classification rates of the rest of the dataset show a different result compared to the rates on the small validation set. Both mature and juvenile lobsters have a relatively low classification rate of about 50%. This was unexpected based on the validation results which especially show juvenile lobsters being have strong results. Further, there is very little difference between the graph and label methods which was not the case during validation. Interestingly, the reason for the poor classification is different for both categories. For juvenile lobsters, it is because a large number of images were classified incorrectly. On the other hand, for mature lobsters, a large percentage was unclassified while only a small percentage was classified incorrectly. This clearly shows how the images chosen for annotation throughout the project were biased to some extent, as the classification during validation got up to 80% accuracy for juvenile lobsters. This suggests how a larger and varied annotated dataset is important for better consistency of results and perhaps better results overall. The pattern of high numbers of unclassified mature lobsters continues in the dataset and greatly affects the overall classification rate as over 30% of all images could not be matched to a graph and therefore could not be classified.

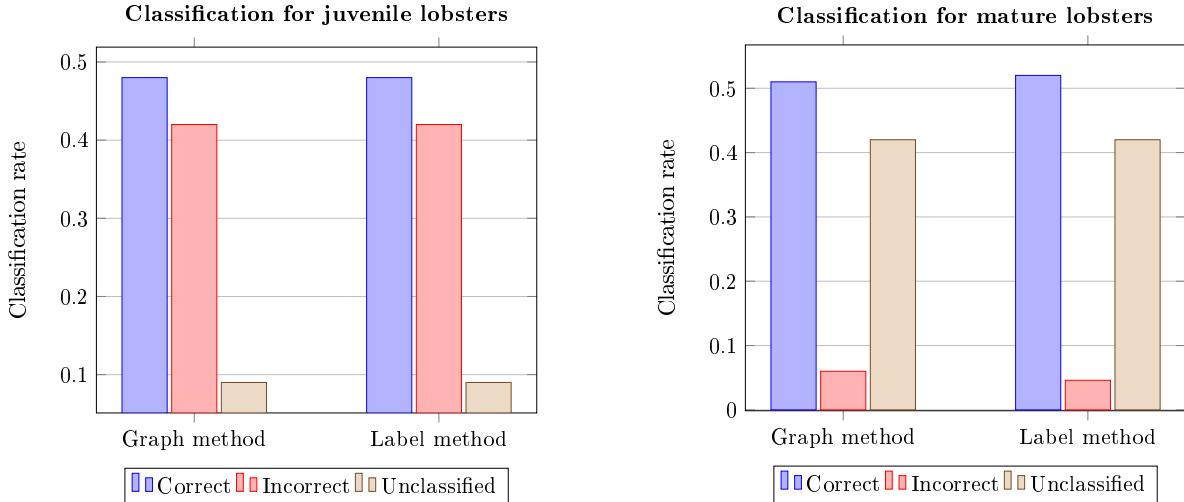


Figure 13: More detailed classification results on the specific lobster categories.

These results are important for finding the areas of weakness in this graph matching approach to correct them for any future work. Having the distinct difference of weaknesses between the juvenile and mature lobsters allows focused evaluation on why each category was stronger and weaker in different aspects.

# 3 Evaluation

## 3.1 Related work

The precision and recall values cannot be easily compared to Abdallah's numbers [1] as the precision and recall of the keypoint identification and labelling was tested in this project, rather than the different classifications in his work. In his analysis, the highest classification accuracy from all his methods was bootstrap aggregation using linear regression with an accuracy of 86.2% while the lowest accuracy of his methods was a linear SVM using principal component analysis which got an accuracy of 76.5%.

In this project, the overall accuracy of juvenile/mature category categorisation is only about 50%. This is a low result, however, it is interesting to note that about 30% of the classifications were unsuccessful. This shows the weakness of this approach coming from the inability to classify rather than the accuracy of the classification itself. The inability comes from when a graph could not be built out of the image. This can be due to the various thresholds and probabilities in the stages leading up to the graph creation filtering too many relevant keypoints or subgraphs away. It can also be due to issues in detection of the images as it more specifically affects the third set of images in the dataset. Additionally, if a larger attributed graph dataset could be built, more optimisations done to allow for more lenient thresholds and the percentage of unclassified images decreased, then the use of graph matching can be further explored with more promising results. Furthermore, it should be noted that the issue of being unable to classify the images comes mostly from mature lobsters.

The category classification of juvenile lobsters is also surprisingly poor for a different reason: many juvenile lobsters are classified incorrectly as mature lobsters. This indicates a potentially different issue compared to the problem of unclassified mature lobsters. It is possible a larger annotated dataset can solve the issue, as an improved model for mature lobsters will lead to decreased incorrect juvenile classification.

## 3.2 Objectives

The project met all of the primary objectives set out in the beginning and explored certain secondary objectives during the design and implementation.

### 3.2.1 Primary objectives

#### **Explore and create suitable graph representations for lobsters:**

At the beginning of the project, Gephi was used to create different configurations of lobster graphs. The configurations were tested and prototyped against the probability model used in the labelling and matching stages to determine suitable representations. Additionally, when keypoint detectors were introduced, the graph representations were solidified by the creation of annotated attributed graphs that corresponded directly to keypoints that were detected.

#### **Match lobster subgraphs with complete graphs using existing software:**

Three software tools were investigated to match example subgraphs with complete graphs. They were examined for their suitability and ease of use for the project. The more powerful tool (nauty) was discovered to be more difficult and less suitable for the problem that needed to be solved. The final tool that was chosen (GraphGrep) was able to take a list of query subgraphs and return a list of matched results based on a database of pre-defined complete graphs.

#### **Automatically detect interest points from images:**

The use of different computer vision algorithms for corner and interest point detection was explored and compared. The corner detectors were not appropriate for the problem as expected, while the point detectors

were able to give meaningful results. All the points detected are then passed through two different filters and a probabilistic model to be labelled after body parts such as claw, tail and head.

**Evaluate this method of graph matching on lobsters against existing dataset:**

After being able to produce a matched graph from an image, the performance was measured by calculating metrics such as precision, recall and F1 score. These scores were analysed to show the strength and weakness of different stages of the project's approach. Finally, classification accuracy was compared to the results that accompanied the original dataset to show how this method performs in comparison. Weakness were identified in specific aspects of the methods and models, which lets further work focus specifically in the areas of reducing unclassified mature lobster images and reducing incorrect classification of juvenile lobster images.

### **3.2.2 Secondary objectives**

**Extend existing or come up with new algorithms for creating lobster graphs from images:**

The entire methodology to go from an image of a lobster to the complete graph is a new approach which combines many existing algorithms and methods together. Although no new algorithm was used in each stage, the resultant methodology is a novel technique combining computer vision, graph matching and probabilistic models to create an attributed lobster graph from images of lobsters.

**Apply this technique on more complex images with noise such as natural lobsters in their environment:**

The application of the technique used in this project on noisy background was explored when investigating the use of colour histograms as a keypoint filter. Further, the use of probabilistic models negated issues with noise as irrelevant keypoints are less likely to be matched. However, a complete evaluation of the methods used in this project on noisy backgrounds was not fully realised.

**Apply this technique for video instead of images to give properties and information on the lobster in real time:**

The use of methods in this project for real time video were not explored due to both time constraints of the project. This leaves scope for further work to be done in this area, which would result in a more concrete software artefact based on this research that can have significant real world applications.

# 4 Conclusion

## 4.1 Discussion

This project has studied the usage of attributed graphs for lobster body part detection and lobster categorisation. The ability to categorise lobsters automatically is important to help fisheries and scientists gather data for further study and management.

By using software techniques compared to traditional manual methods, the effort required is reduced, greatly enhancing efficiency. Nevertheless, errors and accuracy issues can occur. Most importantly, if the body keypoint is unable to be identified from the image, the remaining graph matching process is unable to be built. Thus, more robust methods for graph matching have to be designed to get around this difficult problem. Furthermore, the variety of methods and thresholds can each have different applications, as different combinations work better in different cases. For example, low label thresholds give strong performance for keypoint identification, but higher thresholds are needed for specific labelling like the head and arm labels. Even when a complete graph is matched to the image of a lobster, as the F1 score suggests, it is typically far from being completely accurate. Different use cases for the graph matching approach studied would care about different aspects of the accuracy. For example, if the total length of the lobster is required, only the head, body and tail keypoints are relevant as the length can be calculated by the distance between these keypoints. The methods and models can then be adjusted based on these needs, though further study would be required to deduce exactly how the methods and thresholds should be adjusted.

It would be interesting to see the application of graph matching on other creatures and species. As it as seen successful application for humans [13, 27] and now for lobsters, it is highly likely the approach can be applied towards other animals, especially those with distinct body features that can be captured from images. The advantage of using graph matching is the additional information due to the connections between nodes. The application of this approach on other animals would again come from any need to monitor or measure properties about them where the manual monitoring process is difficult. The graph matching approach would be able to give information on size in most cases if body parts or features can be found reliably. The ability to find any more data based solely from matched graphs is unlikely, but other techniques can be applied on specific nodes or edges after a graph matching step.

## 4.2 Future work

### 4.2.1 Model improvements

As explained in the results and evaluation sections, there are specific issues that can be improved to improve the overall classification rate of mature and juvenile lobsters. It is most likely that a larger annotated dataset will have a large impact on performance, as the results showed the small annotated dataset used was biased towards well classified juvenile lobsters.

### 4.2.2 Other measurements

There are a few other measurements that could have been made and evaluated using the graph output after matching. Most notably, the size of the lobster can be estimated by converting the size of the labelled keypoints to an actual size in metric units. The methodology used provides support for further work in this area, as SIFT is scale invariant, so image sizes are not an issue as long as they are mapped accordingly to actual sizes. These measurements would allow further classification evaluations to be completed such as sex categorisation based on the width of the abdomen and carapace [15] as well as a more comprehensive mature/juvenile categorisation using the carapace length. Furthermore, details involving the lengths and distances between different body parts using the weighted edges can be used to explore if other elements such

as health or age correlate to different sizes of lobsters. This can help facilitate further research in marine biology as features are able to be automatically detected, allowing for more efficient research.

#### 4.2.3 Real time

A secondary objective that was not explored in this project is the application of the methodology to real time video rather than just images. Achieving speed up to allow for processing the lobster graph matching problem in real time allows for greater real world applications involving live cameras on fishing vessels. For example, the use of cameras aboard fishing vessels has already been explored in [15], though manual work was used to process the video feed. An extension to this project to allow for real time feedback based on matching results could help greatly. Currently, the time taken to go from an image to a matched graph takes up to a few seconds per image, with lower thresholds taking even longer. To apply this to video with real time feedback, more work will need to be done on optimising the probabilistic model, especially when it comes to the creation of permutations before matching. An example of an optimisation that could be done is the observation that each subgraph permutation must contain a node labelled body due to the the lobster graph representation. Small optimisations such as this can go a long way to improving the methods in this project to allow for the process to be done in real time.

### 4.3 Summary

In conclusion, this project looks into a graph matching approach to detect and label individual body parts of a lobster. Keypoints are detected with computer vision algorithms and are combined using probabilistic models and subgraph matching to build a complete lobster graph from an image. It was discovered that the approach was both feasible and accurate in most cases for detection and labelling of specific body parts. However, in the evaluation of applying the method to lobster classifications, issues with low accuracy for suggests more work must be done, particularly with the use of larger annotated datasets and better defined lobster models.

# References

- [1] A. A. Abdallah. "Machine Learning with Lobsters". MSc Thesis. University of St Andrews, 2017.
- [2] M. Bastian, S. Heymann, and M. Jacomy. "Gephi: An Open Source Software for Exploring and Manipulating Networks". In: (2009). URL: <http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154>.
- [3] V. Bonnici et al. "APPAGATO: an APproximate PArallel and stochastic GrAph querying TOOl for biological networks". In: *Bioinformatics* Vol 32, Issue 14 (2016).
- [4] G. Bradski. "The OpenCV Library". In: *Dr. Dobb's Journal of Software Tools* (2000).
- [5] J. Chen et al. "The Comparison and Application of Corner Detection Algorithms". In: *Journal of Multimedia* Vol. 4 Issue 6 (), pp. 435–441.
- [6] M.-C. Chuang, J.-N. Hwang, and K. Williams. "A Feature Learning and Object Recognition Framework for Underwater Fish Images". In: *IEEE Transactions on Image Processing* Vol 25, Issue 4 (2016).
- [7] S. A. Cook. "The Complexity of Theorem-proving Procedures". In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing*. STOC '71. Shaker Heights, Ohio, USA: ACM, 1971, pp. 151–158.
- [8] M. Fouad et al. "Automatic Nile Tilapia fish classification approach using machine learning techniques". In: *13th International Conference on Hybrid Intelligent Systems, HIS 2013*. Dec. 2013.
- [9] R. Giugno and D. Shasha. "GraphGrep: A Fast and Universal Method for Querying Graphs". In: *International Conference on Pattern Recognition* (2002).
- [10] J. Hambrey et al. *Evidence Gathering in Support of Sustainable Scottish Inshore Fisheries: Integrating Stock Management Considerations with Market Opportunities in the Scottish Inshore Fisheries Sector - a Pilot Study*. Tech. rep. WP6. 2015.
- [11] M. Hao, H. Yu, and D. Li. "The measurement of Fish Size by Machine Vision - A Review". In: *Computer and Computing Technologies in Agriculture IX: 9th IFIP WG 5.14 International Conference*. Beijing, China, 2015.
- [12] H. Bunke. "On a relation between graph edit distance and maximum common subgraph". In: *Pattern Recognition Letters* (1997).
- [13] T. Hidenori, N. Atsushi, and T. Haruo. "Human Pose Estimation from Volume Data and Topological Graph Database". In: *Computer Vision – ACCV 2007*. Springer Berlin Heidelberg, 2007, pp. 618–627.
- [14] S. Hirai. "Color Filter in SIFT Matching". In: *Proceedings of the 2013 JSME Conference on Robotics and Mechatronics*. Tsukuba, Japan, 2013.
- [15] N. Hold et al. "Video capture of crustacean fisheries data as an alternative to on-board observers". In: *ICES Journal of Marine Science* 72.6 (2015), pp. 1811–1821.
- [16] E. Karami, S. Prasad, and M. S. Shehata. "Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images". In: *Computer Vision and Pattern Recognition* abs/1710.02726 (2015).
- [17] D. Koutra et al. "Algorithms for Graph Similarity and Subgraph Matching". In: (Apr. 2018).
- [18] J. Lehtosalo and G. van Rossum. *mypy*. URL: <http://www.mypy-lang.org>.
- [19] D. G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". In: *Int. J. Comput. Vision* 60.2 (Nov. 2004), pp. 91–110. ISSN: 0920-5691.
- [20] F. Martin-Rodriguez and M. Barral-Martinez. "iObserver: Species Recognition via Computer Vision". In: *7th International Workshop on Marine Technology*. SARTI, 2016, pp. 37–39.

- [21] B. D. McKay and A. Piperno. “Practical graph isomorphism, {II}”. In: *Journal of Symbolic Computation* 60 (2014), pp. 94–112. ISSN: 0747-7171. URL: <http://www.sciencedirect.com/science/article/pii/S0747717113001193>.
- [22] F. C. Monteiro. “Automatic Cattle Identification Using Graph Matching Based on Local Invariant Features”. In: *Proceedings of the 13th International Conference on Image Analysis and Recognition*. Povoa de Varzim, Portugal, 2016, pp. 792–800.
- [23] C. F. Olson and S. Zhang. “Keypoint Recognition with Histograms of Normalized Colors”. In: *2016 13th Conference on Computer and Robot Vision (CRV)*. Victoria, BC, Canada: IEEE.
- [24] M.-J. Rochet, T. Catchpole, and S. Cadrian. “Bycatch and discards: from improved knowledge to mitigation programmes”. In: *ICES Journal of Marine Science* 71(5) (2014).
- [25] A. Rodriguez et al. “Fish Monitoring and Sizing Using Computer Vision”. In: *Bioinspired Computation in Artificial Systems*. 2015, pp. 419–428.
- [26] H. Sharma et al. “A review of graph-based methods for image analysis in digital histopathology”. In: *Diagnostic Pathology* (2015).
- [27] M. Straka et al. “Skeletal Graph Based Human Pose Estimation in Real-Time”. In: *Proceedings of the British Machine Vision Conference*. 2011, pp. 69.1–69.12.
- [28] opencv dev team. *Feature Detection and Description*. URL: [https://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_feature2d/py\\_table\\_of\\_contents\\_feature2d/py\\_table\\_of\\_contents\\_feature2d.html](https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_table_of_contents_feature2d/py_table_of_contents_feature2d.html).
- [29] *The DOT Language*. URL: <https://www.graphviz.org/doc/info/lang.html>.

# Appendices

## Appendix A Keypoint detection on annotated dataset using different algorithms

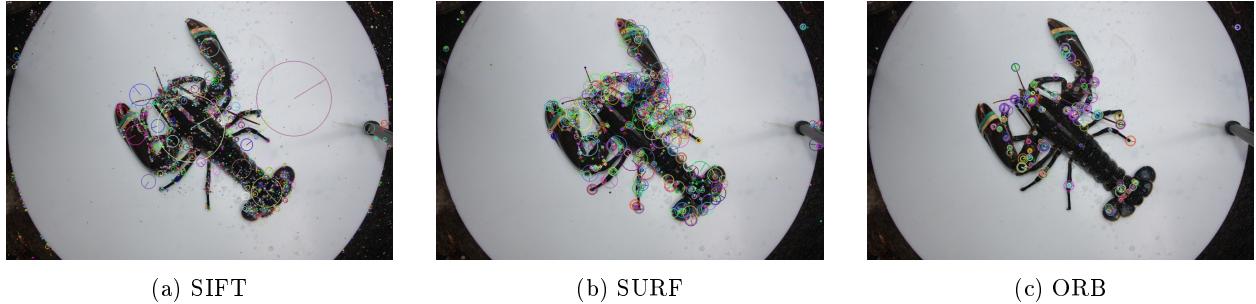


Figure A.1: Detection algorithms on image 1380

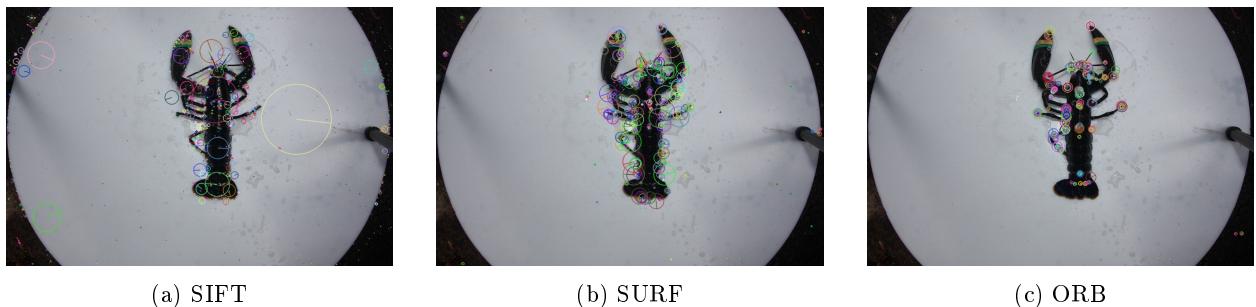


Figure A.2: Detection algorithms on image 1398

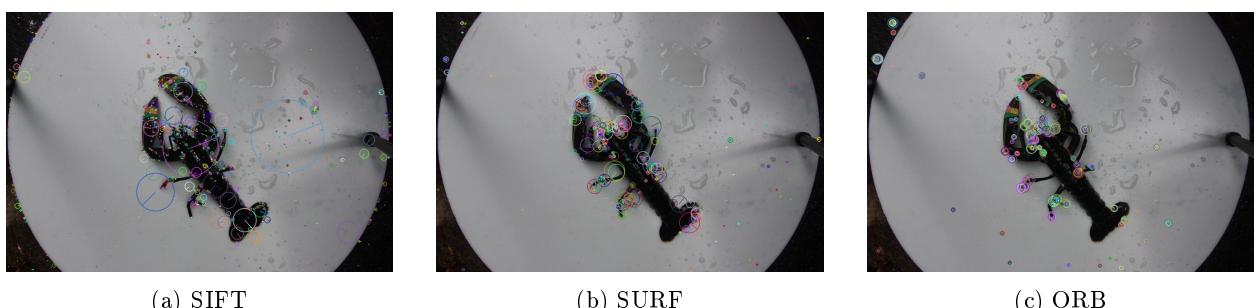


Figure A.3: Detection algorithms on image 1424

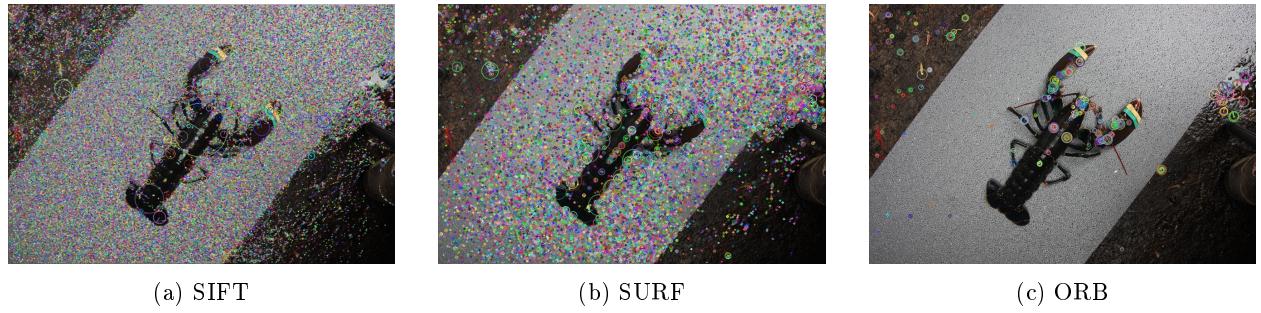


Figure A.4: Detection algorithms on image 1487

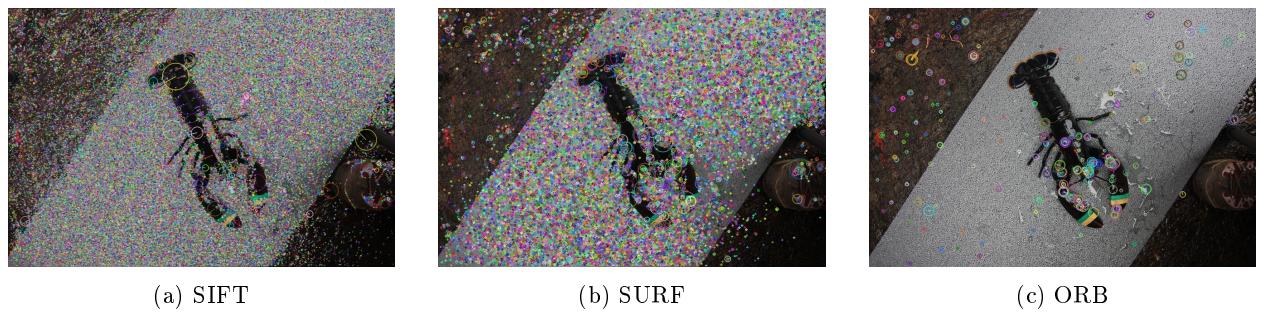


Figure A.5: Detection algorithms on image 1507

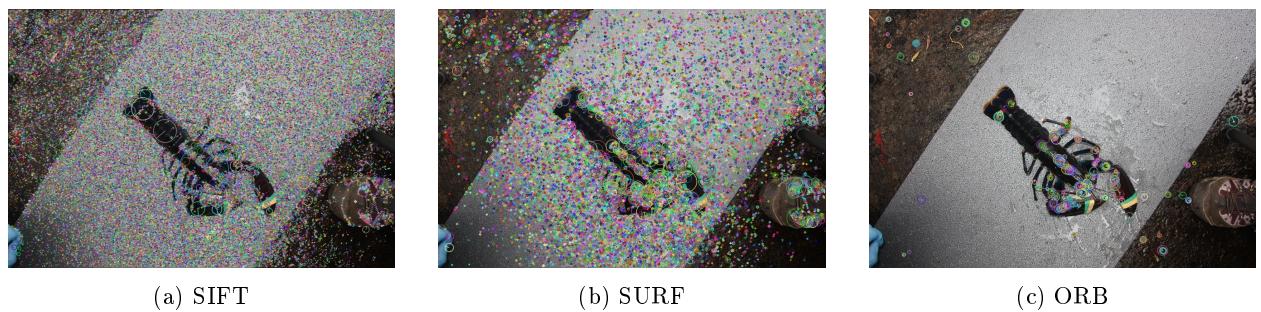


Figure A.6: Detection algorithms on image 1516

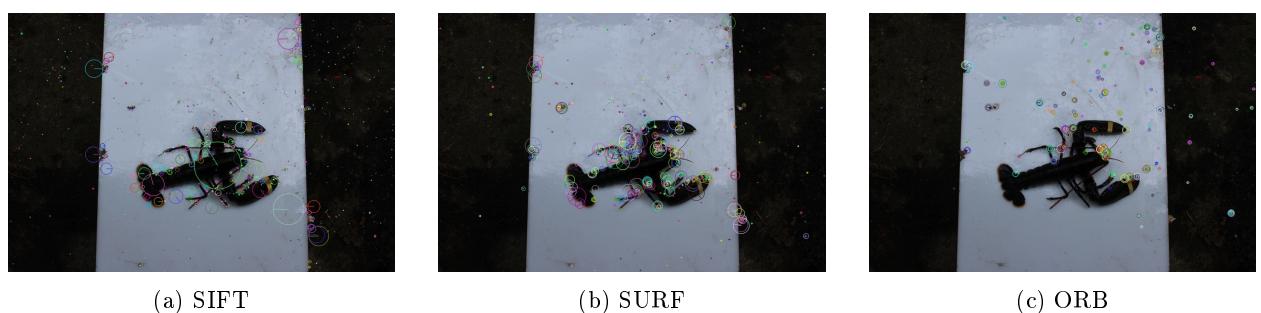


Figure A.7: Detection algorithms on image 4721

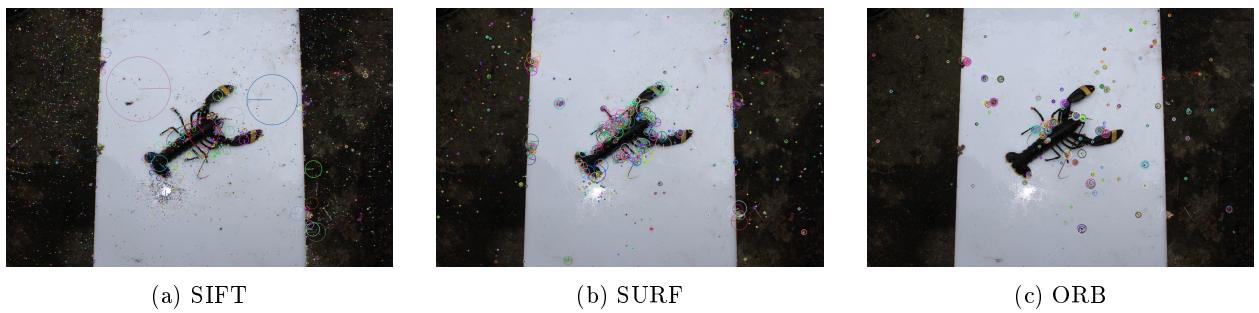


Figure A.8: Detection algorithms on image 4728

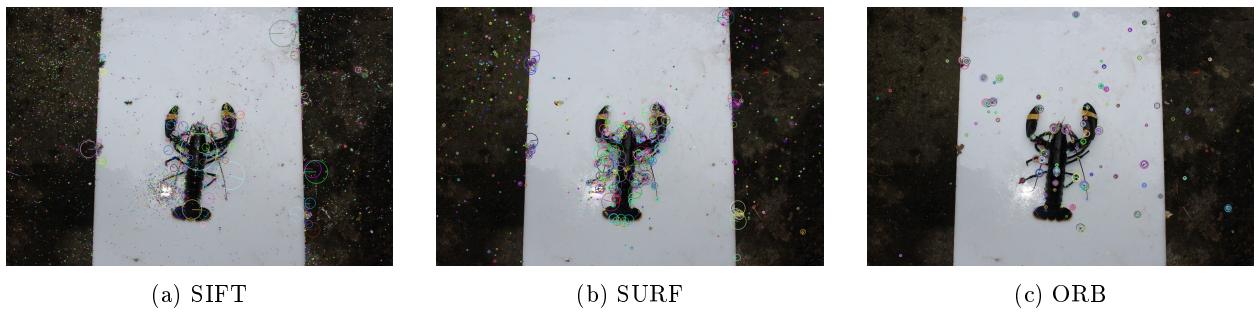


Figure A.9: Detection algorithms on image 4737

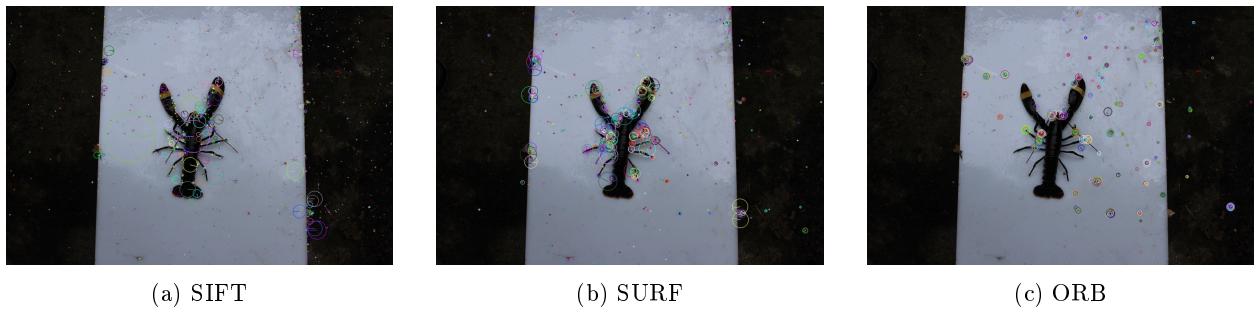


Figure A.10: Detection algorithms on image 4772

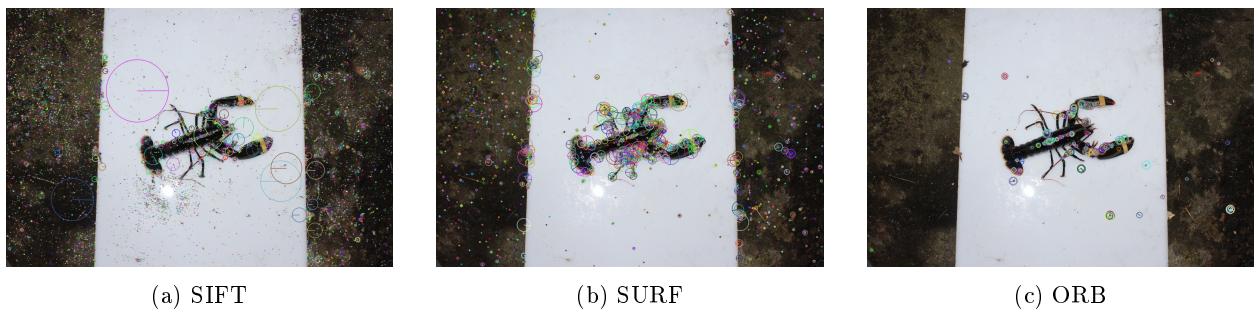


Figure A.11: Detection algorithms on image 4785



(a) SIFT



(b) SURF



(c) ORB

Figure A.12: Detection algorithms on image 4799



(a) SIFT



(b) SURF



(c) ORB

Figure A.13: Detection algorithms on image 4805



(a) SIFT



(b) SURF



(c) ORB

Figure A.14: Detection algorithms on image 4825



(a) SIFT



(b) SURF



(c) ORB

Figure A.15: Detection algorithms on image 4836



(a) SIFT

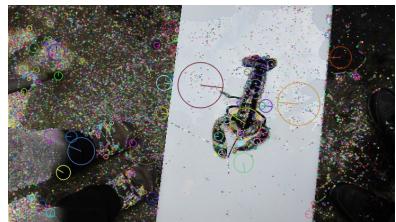


(b) SURF



(c) ORB

Figure A.16: Detection algorithms on image 4844



(a) SIFT



(b) SURF



(c) ORB

Figure A.17: Detection algorithms on image 5252



(a) SIFT



(b) SURF



(c) ORB

Figure A.18: Detection algorithms on image 5298



(a) SIFT



(b) SURF



(c) ORB

Figure A.19: Detection algorithms on image 5348

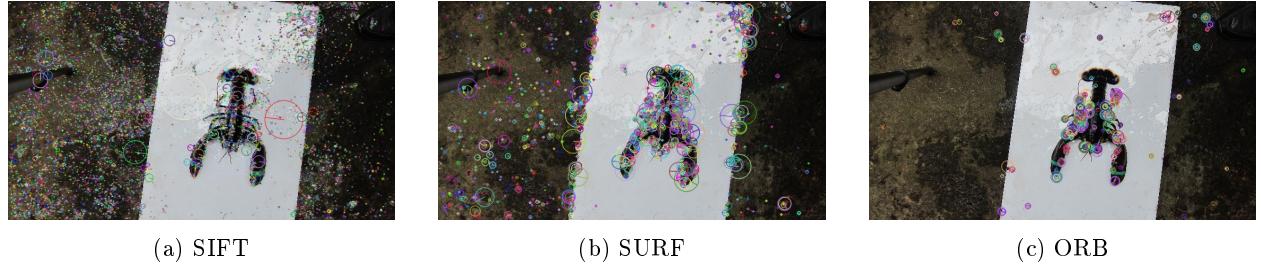


Figure A.20: Detection algorithms on image 5414

## Appendix B Precision and recall results for keypoint labelling

## Appendix C F1 score results for keypoint labelling

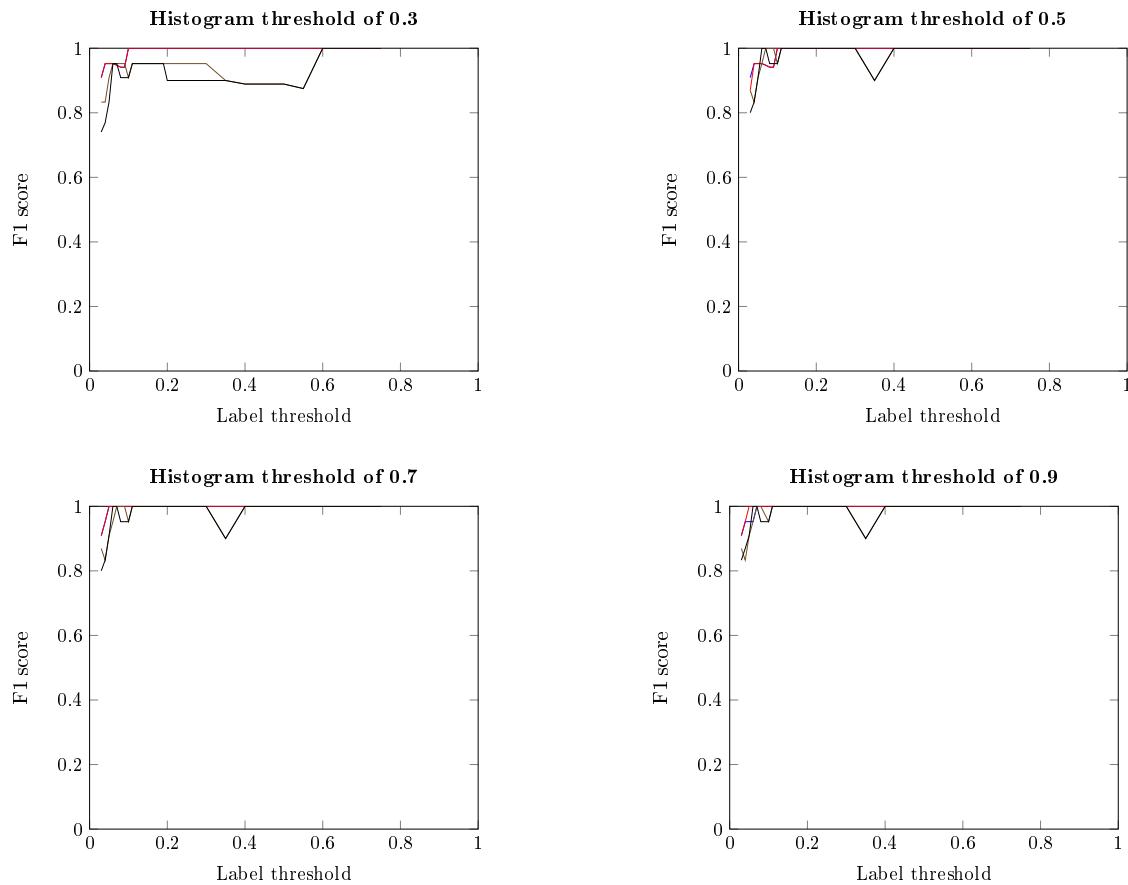


Figure C.1: F1 score graphs with mature model for *body* label

- |   |                                   |
|---|-----------------------------------|
| — | Graph method on mature lobsters   |
| — | Label method on mature lobsters   |
| — | Graph method on juvenile lobsters |
| — | Label method on juvenile lobsters |

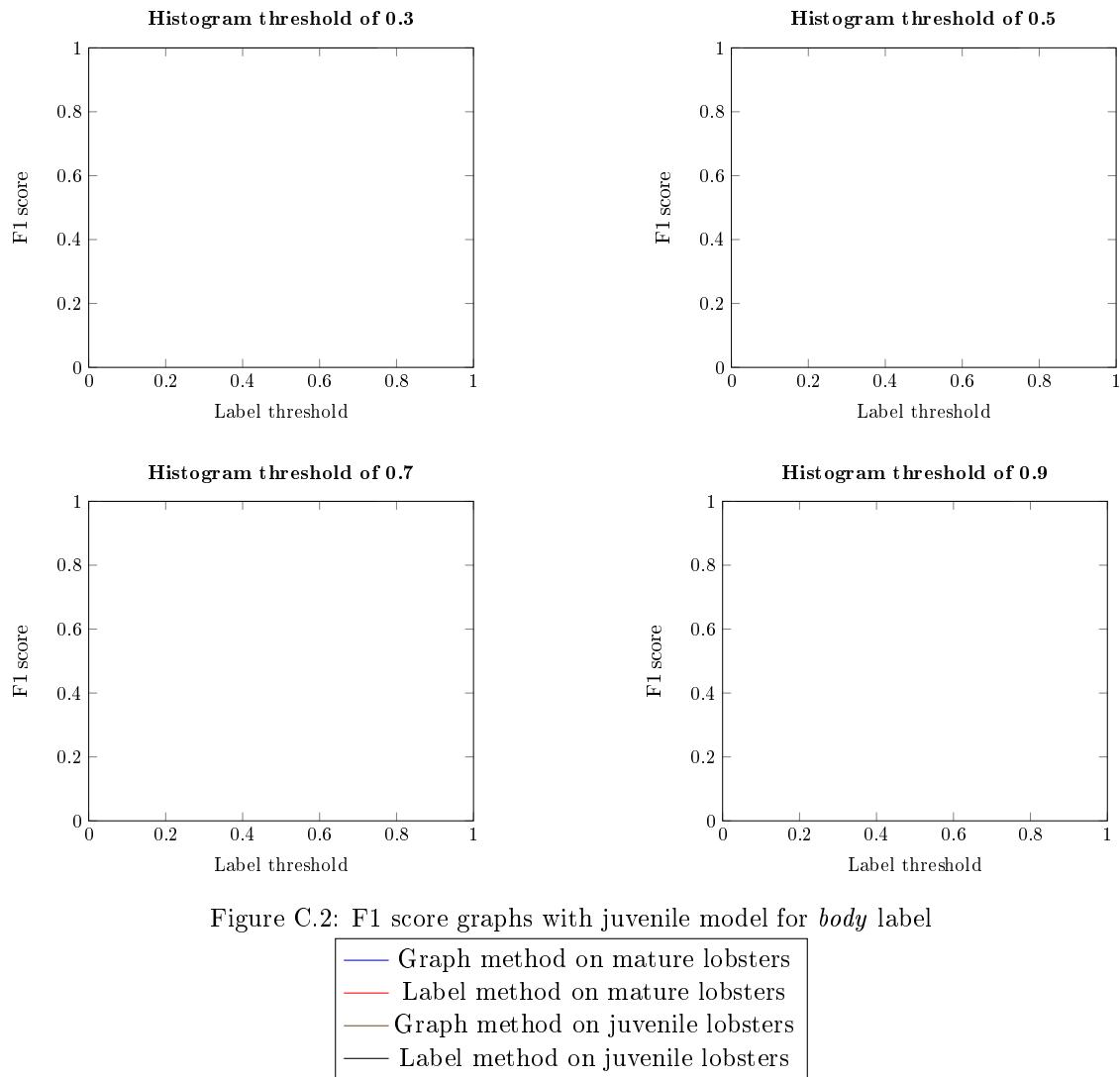


Figure C.2: F1 score graphs with juvenile model for *body* label

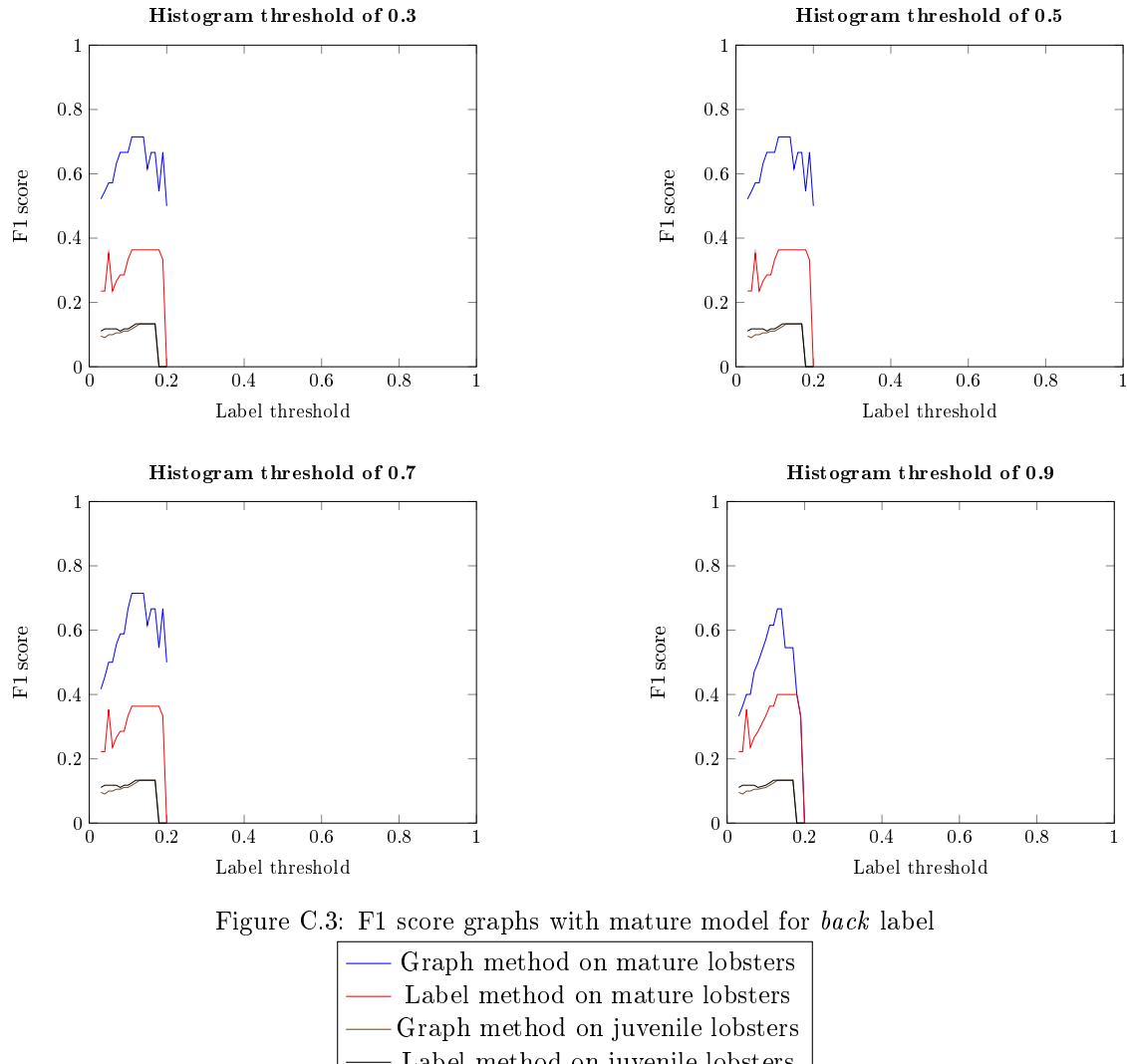


Figure C.3: F1 score graphs with mature model for *back* label

- |   |                                   |
|---|-----------------------------------|
| — | Graph method on mature lobsters   |
| — | Label method on mature lobsters   |
| — | Graph method on juvenile lobsters |
| — | Label method on juvenile lobsters |

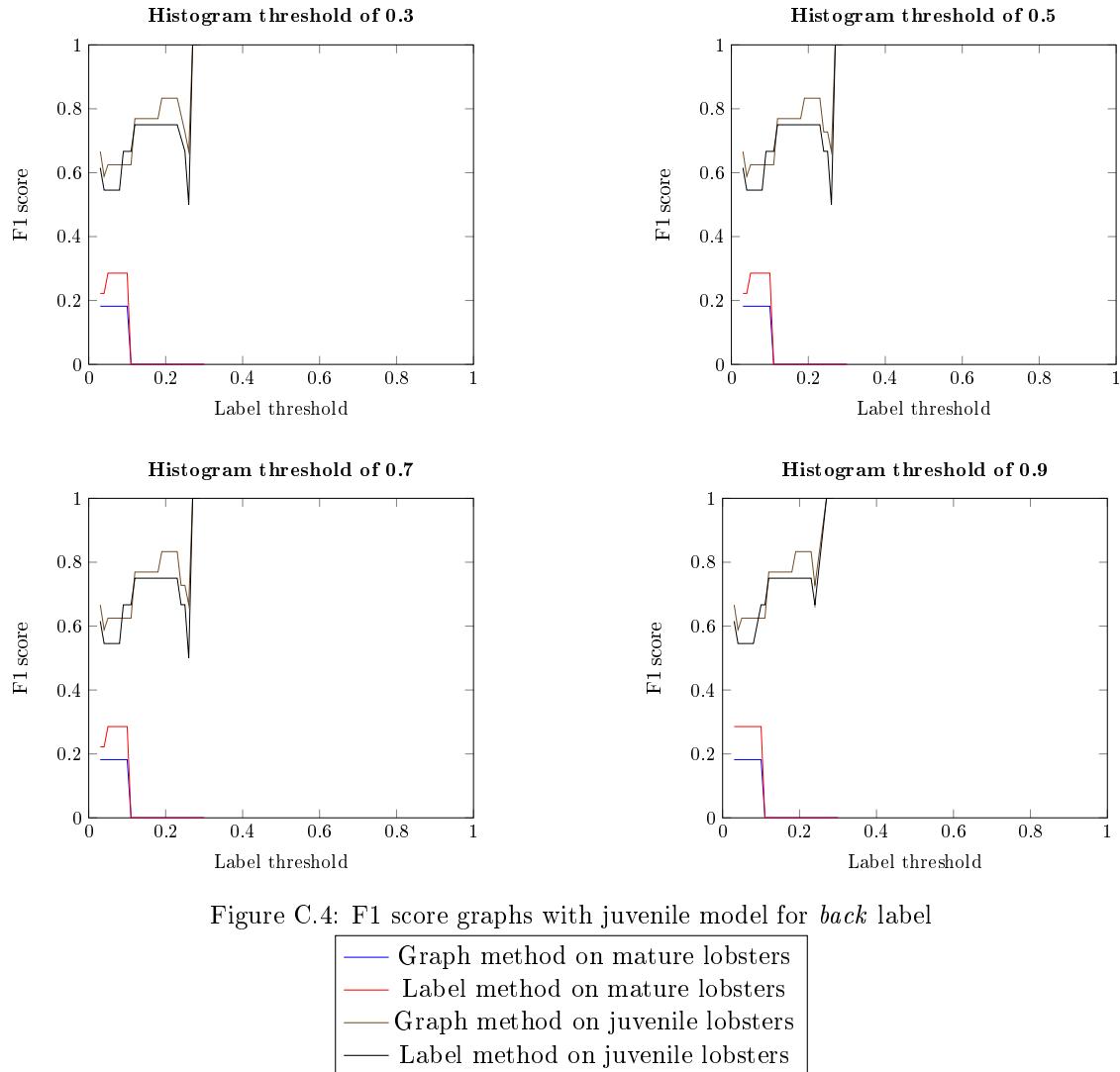


Figure C.4: F1 score graphs with juvenile model for *back* label

- |   |                                   |
|---|-----------------------------------|
| — | Graph method on mature lobsters   |
| — | Label method on mature lobsters   |
| — | Graph method on juvenile lobsters |
| — | Label method on juvenile lobsters |

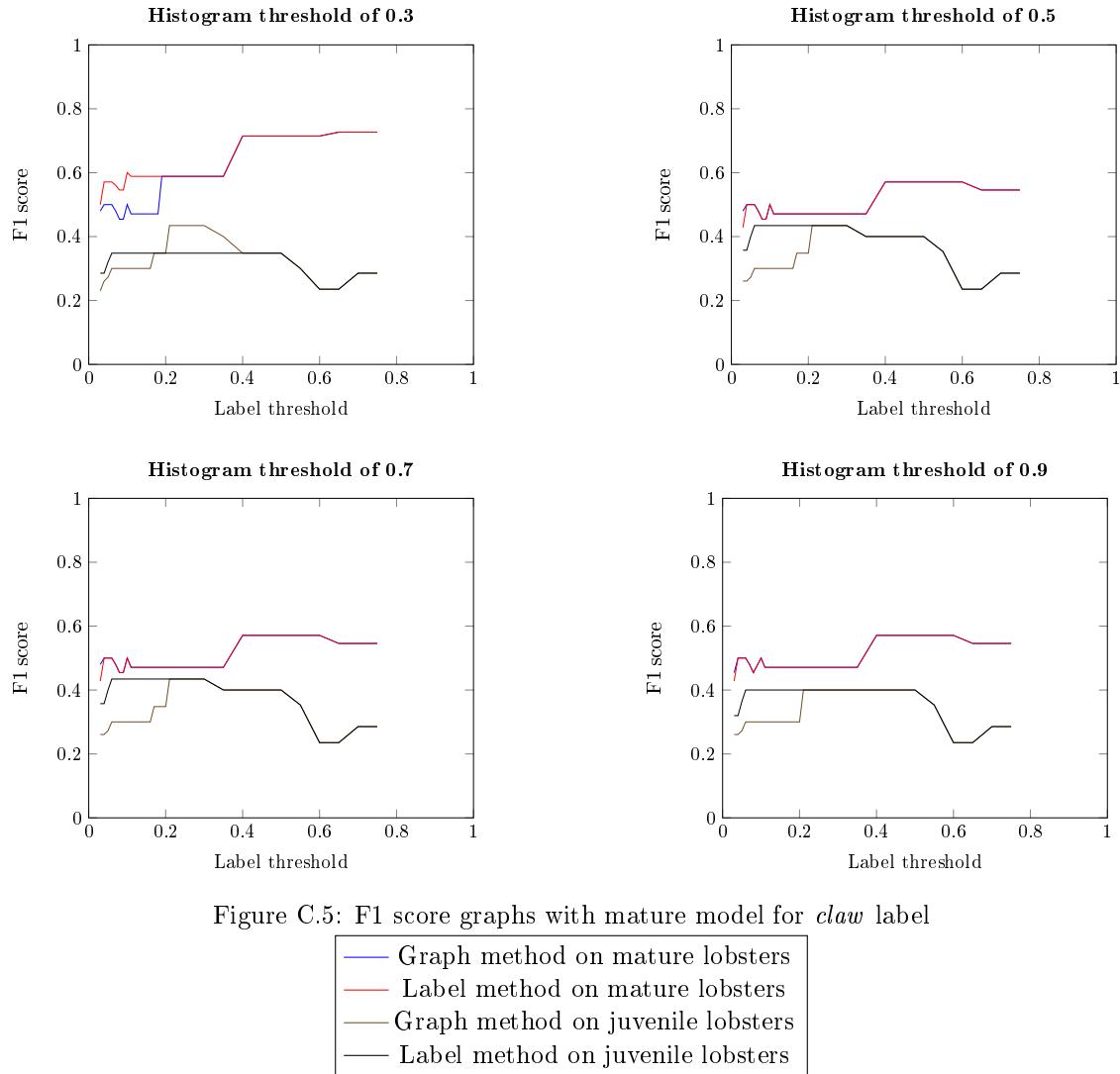


Figure C.5: F1 score graphs with mature model for *claw* label

- |     |                                   |
|-----|-----------------------------------|
| —   | Graph method on mature lobsters   |
| —   | Label method on mature lobsters   |
| - - | Graph method on juvenile lobsters |
| - - | Label method on juvenile lobsters |

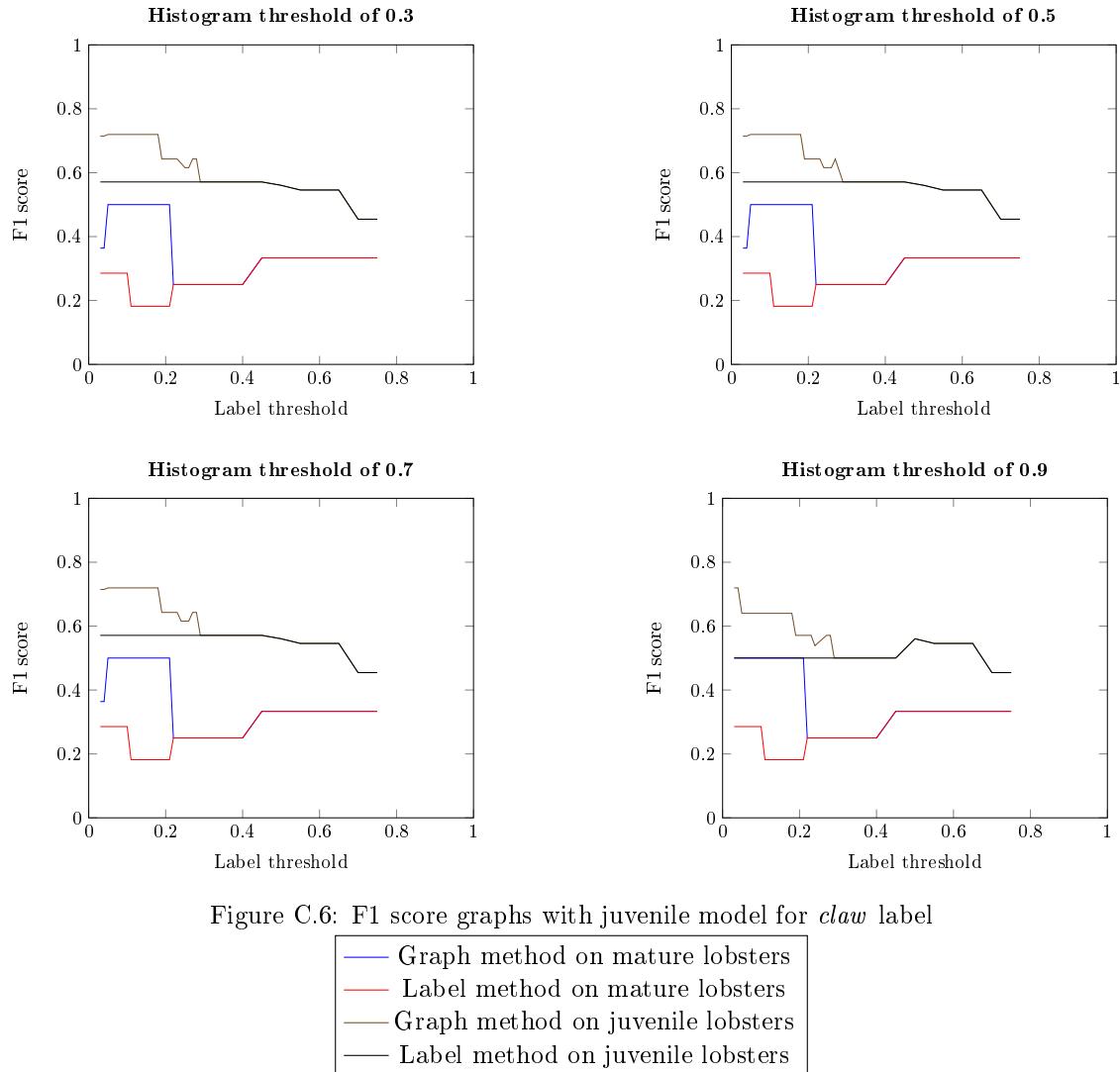


Figure C.6: F1 score graphs with juvenile model for *claw* label

- Graph method on mature lobsters
- Label method on mature lobsters
- Graph method on juvenile lobsters
- Label method on juvenile lobsters

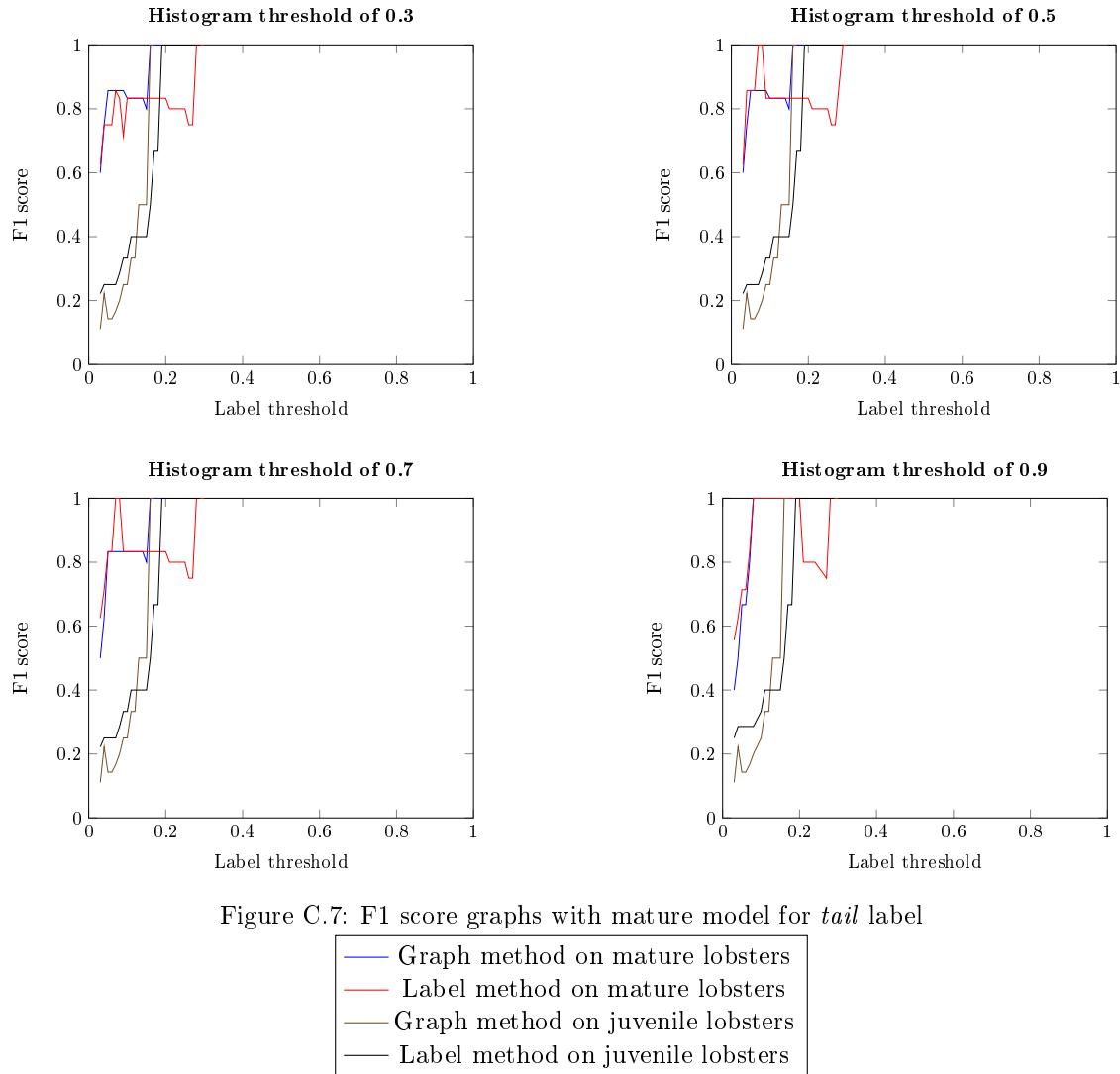


Figure C.7: F1 score graphs with mature model for *tail* label

- |     |                                   |
|-----|-----------------------------------|
| —   | Graph method on mature lobsters   |
| —   | Label method on mature lobsters   |
| - - | Graph method on juvenile lobsters |
| - - | Label method on juvenile lobsters |

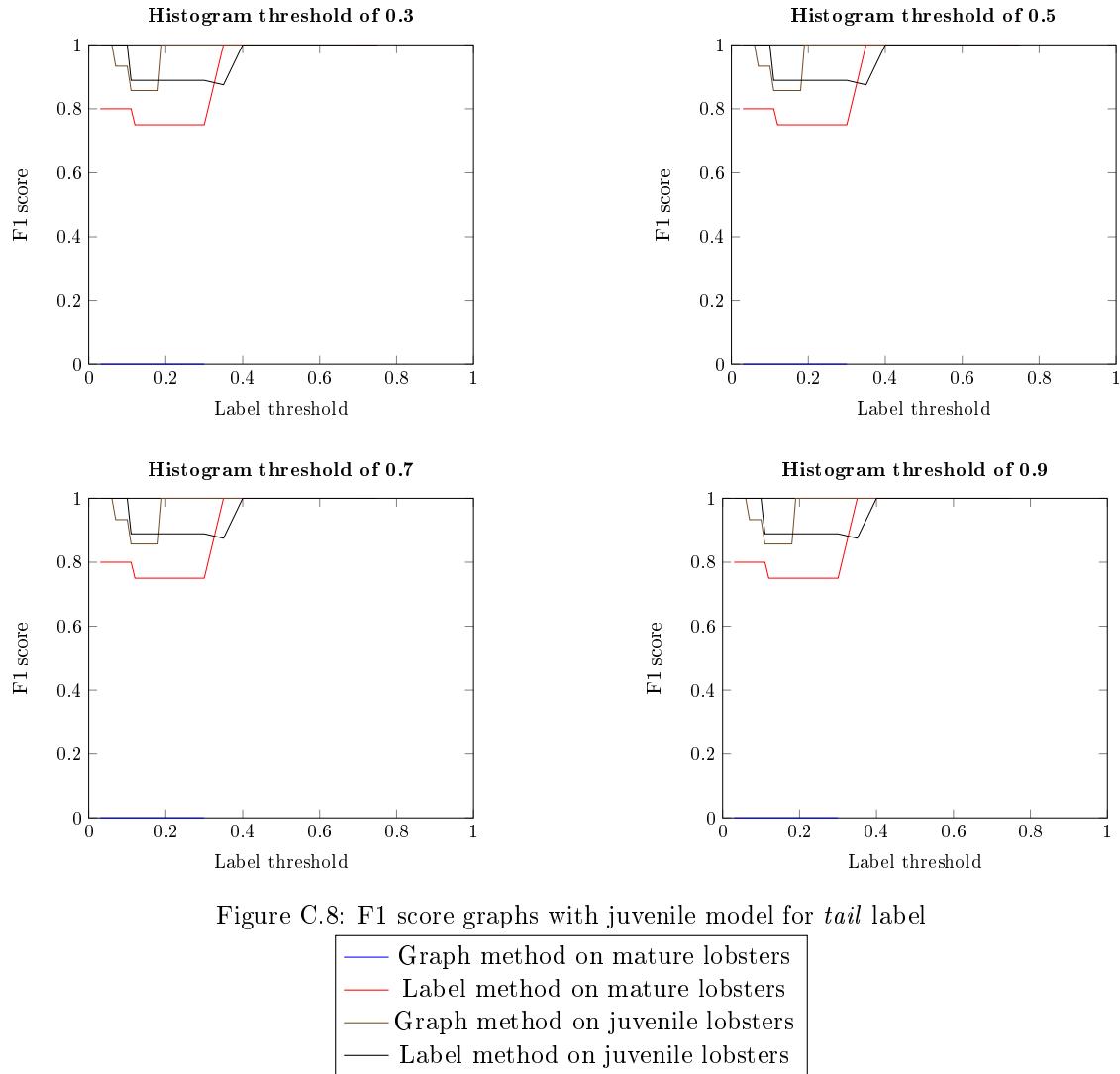


Figure C.8: F1 score graphs with juvenile model for *tail* label