7 Detected segment matching

Haar Classifier Cascades were developed for face detection (Viola & Jones, 2001b) and have otherwise been applied to finding similar mostly-rigid objects. Seahorses are much more flexible. Even annotating whole seahorses with consistent training regions seemed prone to error, so no such attempt was made.

Instead, separate seahorse head and body training sets were constructed, as discussed in section 5.2.2. Classifiers trained from these sets detected segments as described in sections 5.4.3 and 6.4.5, but the segment detections could not give the position or pose of complete seahorses. Therefore, the system needed to match detected bodies to detected heads. The approach used needed three components: cascades for detecting the segments, a formula giving the cost of linking a given head and body together, and an algorithm to choose segments to join given their cost. As with most problems in object detection, there is probably no single best accuracy: the matching algorithm should generate an ROC curve. It may then offer a choice between high true positives/high false positives and low true positives/low false positives, as discussed in section 2.7.3.

Segment matching combines information from both head and body detections, so may be able to compensate for any weaknesses in either of the individual detectors to provide a more robust seahorse detector. Seahorse pose detection, mentioned in section 4.3, also requires that both segments be matched.

7.1 Cascades

The seahorse segment detectors trained in chapter 5 were used once again. They were then used to make both binary detections and to create confidence maps as described in chapter 6. Following the results in section 5.4.5, the rotation was in 15° steps; this rotation was on the images, not the cascades, in accordance with the findings in section 5.4.6. For binary detection, the results in section 5.4.3 suggested using the head detection cascade trained on 5° random angle ranges and body detection cascade trained on 10° ranges. For confidence mapping the head and body cascades were those
trained on 20° and 15° angle ranges respectively, following the results in section 6.4.6.

It must be noted that these matching algorithms will at best correctly detect as many full seahorses as had both parts found by the original detectors. For binary detection, this is 519; for confidence mapping it is 614. Even these counts are likely to be unattainable, as high true positive rates are only obtained at very low confidence thresholds, which in turn let through many false positives.

7.2 Matching formulae

The intention here is to create a formula giving the ‘cost’ or ‘difficulty’ of matching a given head to a given body. Its output should be positive: close to 0 for a likely match, and increasing for less likely matches.

The inputs to this formula may be derived from pairs of potentially matched segments, such as the two in fig. 7.1. Information such as that listed below then becomes available.

![Figure 7.1: Properties of seahorse segment detections](image-url)
Chapter 7 Detected segment matching

$c_h$, the confidence of the head segment detection (either the number of neighbours or the confidence map value)

$c_b$, the confidence of the body segment detection (as above)

$l_h$, the length of the head segment in pixels,

$l_b$, the length of the body segment in pixels,

$s$, the separation in pixels between the top of the body segment and the bottom of the head segment

$\theta$, the angle difference between the segments

Given the values above and information about the annotations made during training, further values were derived:

$s_n$, the normalised separation $= \frac{s}{\sqrt{l_h \times l_b}}$

$r$, the ratio between the segment lengths $= \frac{l_b}{l_h}$

$\Delta r$, the difference from the mean $r$ of the training annotations $= |r - 1.66|$  

$\Delta \theta$, the difference the mean $\theta$ of the training annotations $= |\theta - 76^\circ|$  

Preliminary tests showed that the separation was better transformed to $s_n^2$, and the confidences to $\frac{1}{\sqrt{c_h + 1}}$ and $\frac{1}{\sqrt{c_b + 1}}$.

7.2.1 Designed matching

The first matching formula used was manually constructed from the properties above, based on their apparent importance. It multiplied out the properties, although it guaranteed that each term in the cost formula would be at least 1:

$$cost_{designed} = (4 \times s_n^2 + 1) \times (\Delta r + 1) \times \left(\frac{\Delta \theta}{90} + 1\right) \times \frac{1}{\sqrt{c_h + 1}} \times \frac{1}{\sqrt{c_b + 1}}$$

This formula will return low costs if $s_n$ is low (so the segments are nearby), $\Delta r$ is low (the length of the body is about 1.66 times the length of the head) $\Delta \theta$ is low (the angle between head and body is about $76^\circ$) and $c_h$ and $c_b$ are high (the head and body detections have high confidence).
7.2.2 Learnt matching

Formulae were also learnt with the aid of WEKA, the machine learning environment described in section 2.3.5. A dataset was built from every pair of head and body detections left on the training images after running the binary cascade and merging neighbours. The attribute values were \( s_n^2, \Delta r, \Delta\theta, \frac{1}{\sqrt{c_h+1}} \) and \( \frac{1}{\sqrt{c_b+1}} \); the class was 0 if the annotations showed the segments were correct detections of the same seahorse and 1 if they were not. The WEKA ‘Logistic’ classifier learner was then run on this dataset; it chose the formula below.

\[
\text{cost}_{\text{binary}} = 12.30s_n^2 + 2.62\Delta r + 0.023\Delta\theta + \frac{6.47}{\sqrt{c_h+1}} + \frac{8.13}{\sqrt{c_b+1}}
\]

A similar formula was learnt from detections made by confidence mapping.

\[
\text{cost}_{\text{confidence}} = 9.37s_n^2 + 1.32\Delta r + 0.024\Delta\theta + \frac{19.39}{\sqrt{c_h+1}} + \frac{16.28}{\sqrt{c_b+1}}
\]

The learnt formulae also had negative intercept constants; these were discarded. Constants make no difference to the matching algorithms, and negative constants could lead to negative costs – which might confuse the matching algorithm. Since all the coefficients are positive, the formulae learnt will return low costs if \( s_n, \Delta r \) and \( \Delta\theta \) are low while \( c_h \) and \( c_b \) are high. This is consistent with the expected behaviour of each property and the way they were used in the designed matching formula.
7.3 Matching algorithms

An algorithm is then needed to link segments together with the aid of the matching cost formulae listed above. Two such matching algorithms are considered here: a simple greedy algorithm and a simple closest match algorithm.

The Munkres Algorithm (Munkres, 1957), a variation of the Hungarian Algorithm for minimising the total cost of assignment (Kuhn, 1955) was also tested; its performance was poor and it was found to dismantle ‘good’ matchings that indeed represented seahorses in order to connect the parts to other segments which were false positives to begin with. This suggests that the sum of costs is not an appropriate measure to optimise, and the simpler matching algorithms are sufficient.

7.3.1 Greedy

The greedy matching algorithm repeatedly considers every possible pair of heads and bodies. It repeatedly finds the lowest-cost pairing of a head and body and, if neither item is already part of a pair, joins them. If either segment is already assigned, it ignores both and moves to the next-lowest-cost pair. This has the result that every segment becomes part of at most one object.

7.3.2 Closest match

This algorithm assigns every head to its lowest-cost body and every body to its lowest-cost head. These assignments are made even if the closest part is already assigned.

7.4 Evaluation method

The combination of matching formula and matching algorithm allows ROC curve construction: after matching, choose a cost threshold and only consider matched segments with lower cost than the threshold. These curves may then be compared with those of other matching algorithms and formulae, or with the curves for individual segment detections.
7.5 Results

When the two types of matching cost formula (designed and learnt) and the two matching algorithms (greedy match and closest match) were compared, the combination of learnt cost formula and greedy matching algorithm was most accurate. This was the case for both the binary detection comparison in fig. 7.2(a) and the confidence mapping comparison in fig. 7.2(b).

The best seahorse detections matched from binary detections were more accurate than the individual head detections until they had nearly reached the potential maximum; for low false positive/true positive counts they also improved upon the individual body detections. This may be seen in fig. 7.3(a). The true positive count also came very close to the potential maximum of 519 mentioned in section 7.1. This means that improvement by matching more seahorses is probably infeasible; any changes should attempt to reduce the number of false positives created while making the existing matches.

The matching results based on confidence mapped detections were similar. They were consistently more accurate than individual head detections, and were also better than body detections at low false positive/true positive counts, as shown in fig. 7.3(b).

Fig. 7.4 shows that matching starting with confidence mapped detections was more accurate than matching starting with binary detections. This was expected, as the results in section 6.4.5 show that confidence mapping was more accurate than binary detection for finding the individual segments.

Fig. C.14 contains example images showing whole seahorse detections costs resulting from the binary segment detections in fig. C.11 and the confidence mapped segment detections in fig. C.13.
Figure 7.2: ROC curves for whole seahorse detection with varying matching cost formulae and algorithms
Figure 7.3: ROC curves for whole seahorse detection compared with individual segment detections
This chapter considered ways to create whole seahorse detections out of the head and body segment detections in chapters 5 and 6. Formulae were developed to give the ‘cost’ of matching any given head and body, and two algorithms written to combine them. When compared, the best matching formulae were learnt, and the best of the two matching algorithms made greedy selections.

This combination was usually more accurate than the seahorse head detector, and for low false positive counts was also more accurate than the seahorse body detector. While body detection alone is more accurate when high true positive rates are desired, whole seahorse detection can also provide additional pose information that is not available from body detection alone.

Figure 7.4: ROC curves for whole seahorse detection, varying the segment detection method

7.6 Conclusions