Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 55-31 (2022) 363–368

Motion trajectory estimation of salmon using stereo vision

Trym Anthonsen Nygård * **Jan Henrik Jahren** *
Christian Schellewald ** **Annette Stahl** *

* *Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), O. S, Bragstads plass 2D, 7491 Trondheim, Norway (e-mail:trymany@stud.ntnu.no, jan.h.jahren@ntnu.no, Annette.Stahl@ntnu.no).*
** *SINTEF Ocean AS, Brattørkaia 17C, 7010 Trondheim, Norway (e-mail: Christian.Schellewald@sintef.no).*

Abstract:
A main concern for the aquaculture industry is the fish behaviour and welfare. Motion trajectory analysis of salmon at aquaculture farming sites with respect to certain aquaculture operations aims to provide information about the behaviour and possibly stress level of the farmed salmon and may help to generate a general welfare indicator index. Towards this aim we present an innovative computer vision and machine learning based approach for motion trajectory estimation of salmon. Video footage was recorded with a stereo camera setup. Deep learning based object detection was performed to detect particular features. We focused on tracking the fish eyes and heads as a reliable indicators of the fish's position. Feature matching and subsequent 3D reconstruction was performed to calculate the 3D position of the fish from which trajectories of the fish movement were estimated. Related experiments were conducted at an aquaculture research facility under natural lighting conditions and extracted trajectories allowed a qualitative verification. The developed method was verified using synthetic ground truth data produced with an open source computer graphics software for quantifiable performance metrics.

Keywords: Aquaculture, Underwater localization techniques, Vision, recognition and reconstruction for underwater applications

1. INTRODUCTION

In this study we aim to demonstrate the validity of using cameras to monitor salmon in order to automatically analyse the movement of the fish as part of their behaviour. This study is based on video recordings of an experiment where CO_2 was injected as a stressor for the fish. Modern deep learning based object detection is applied to stereo video footage to identify the head and eye of the individual fish. When features are detected in both the left and right stereo image, they are matched and 3D coordinates are calculated. From this motion trajectories are estimated for the fish during the time it is visible in frame. An example of a generated trajectory next to an image with detections from the experimental data is shown in Figure 1. A methodology is developed for combining deep learning for object detection with stereo vision and feature matching, to estimate 3D motion trajectories of salmon as well as its swimming speed. To enable a quantitative performance measure, a simulated data set with a ground truth is used and the methods are tested on this data set in addition to the qualitative performance measures on the experimental data. In this paper, we first do a brief literature review of Related Works (Section 2) followed by an explanation of our methods (Section 3), afterwards the simulation and experimental setup (Section 4) is introduced briefly before the results are presented (Section 5) and discussed (Section 6) before we conclude (Section 7).

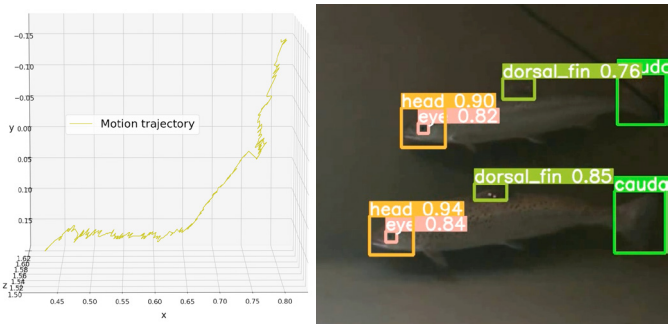

Fig. 1. Trajectory extracted from the eye of a single fish using the proposed methodology. The estimated motion trajectory are shown beside an image with detected features extracted from the recorded video footage.

2. RELATED WORK

Using artificial intelligence and computer vision for aquaculture applications has a long history dating back to 1995 (Newbury et al. (1995), Jovanović et al. (2016), Lien et al.

⋆ This work has received support from the Norwegian Seafood Research Fund FHF (OWITOOLS, Project-number: 901594) and from the Norwegian Research Council (SOUNDWELL, Project-number: 280512).

10.1016/j.ifacol.2022.10.455

(2019), Madshaven et al. (2022)). Recently more work has been reviewed by Yang et al. (2021). Modern machine learning techniques for object detection and tracking, such as Deep Learning which are related to our work are utilized to generate trajectories in 2D over time. They are utilized for behavioural analysis while exposed to a stressor (Xu et al., 2020), for trajectory estimation to evaluate the behaviour of sea cucumbers in an experimental setting (Li et al., 2020), and for a generalised tracker of animals in a group setting (Romero-Ferrero et al., 2019). Related stereo vision approaches for various aquaculture and fisheries applications facilitate classical computer vision methods to segment fish for size estimation (Chuang et al., 2015), or to monitor rail fishing electronically (Huang et al., 2019), or to track fork lengths of farmed tuna, by estimating the 3D coordinate positions from a pair of stereo images with direct linear transform (DLT) (Torisawa et al., 2011).
With regards to previous work, we combine stereo vision and deep learning based object detection to estimate 3D motion trajectories suitable for behavioural analysis of the farmed animals.

3. METHOD

The methodology described in the following section provide the required steps to extract accurate motion trajectories from recorded stereo video footage of salmon.

3.1 Calibration and rectification

Camera calibration is a crucial step in order to extract accurate metric measurements from the stereo images. Camera calibration was performed according to the proposed method by Zhang (2000). During camera calibration the distortion parameters are determined to correct the images for the present distortion (for more details, see Hartley and Zisserman (2000)). The intrinsic camera parameters define the geometry of a pin hole camera model (perspective projection) and were determined by using a checkerboard calibration pattern with known geometry.

Each frame in the stereo videos were also rectified using the calibration data such that the viewing direction would be parallel and orthogonal to the camera baseline. During the rectification process the images gets transformed and warped such that they appear to only have horizontal displacement (Figure 2). Meaning that the displacement along the y-axis would ideally be zero. To ensure that the image planes gets aligned well, it's important to remove optical distortion in the image. The distortion can be removed by using the distortion coefficients that were obtained during the calibration.

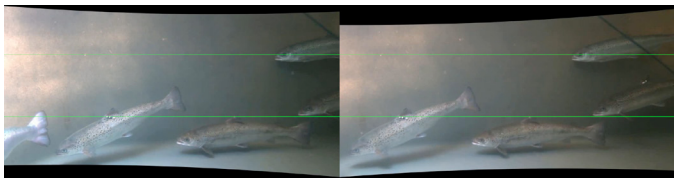

Fig. 2. After rectification corresponding points in both images would be located along the same epipolar lines

3.2 Data preparation

The training data for the neural networks was created by manual annotation of images, both non-rectified and rectified stereo images were annotated using a computer vision annotation tool (Sekachev et al. (2020)). Except for rectification of parts of the training set, no further data pre-processing is applied, prior to training time augmentations.

3.3 Object detection

In order to extract trajectories of the fish movement from the stereo videos, accurate detection and tracking of the fish is essential. There are several candidates for the features of the fish that could be detected and traced such as fins, eyes or head. Considering the accuracy and stability of the detection, the eyes were selected for stereo matching as it empirically provided the most consistent detections (an example of an image with detections can be seen in Figure 1). The head and the dorsal fin can be other possible candidates for tracking. The accuracy measured in mAP (mean average precision) of dorsal fin detections were significantly lower and the head detection bounding boxes varied some more in size. Due to fish gills moving, the bounding box of the head shifts with the breathing of the fish. The caudal fin is not suitable due to sideways movement of the tail. Unlike the experimental data, the head proved to be the most reliable indicator for the synthetic data as the confidence levels made the eye detections less suitable, thus the head is used to generate those trajectories.

For the object detection algorithm, a YOLOv5 architecture is used (Jocher (2021)). The network is trained on self-annotated data as described above using significant amounts of data augmentation. The kinds of augmentation that were used are: Mixup (Zhang et al. (2018)), shearing, scaling, translation, mosaic (Bochkovskiy et al. (2020)), translation, flips (right and left) and HSV (Hue Saturation Value) variation. Training was carried out with up to 500 epochs, with improvements stopping around 300 epochs in terms of mAP which is used to score performance. Stochastic gradient descent with momentum of 0.937 and 3 warm up epochs with 0.8 momentum was used to train the network. Plots of the resulting mAP, training and validation losses (objective, class and box) losses can be seen in Figure 3. The resulting model from this training performed adequately when data quality is high, with a precision at confidence 0.4 of about 80% with a 75% recall overall on the validation data.

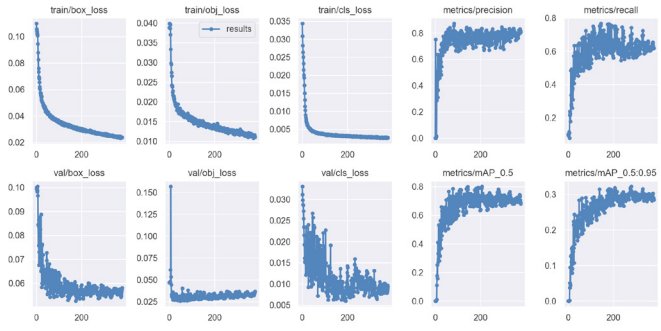

Fig. 3. Graphs of accuracy, precision, recall as well as validation/training losses (box, object and class).

3.4 Multi object tracking

Stereo matching is not trivial when multiple salmon are visible in the stereo images. To simplify the stereo matching, the acquired objected detection data was sorted such that corresponding features in left and right image from the same salmon are grouped together. This allows us to track the same feature in consecutive frames. The tracking was achieved with Bochinski et al. (2017) high speed multi object tracker (MOT). Bochinski's algorithm is sorting the data based on overlapping bounding boxes in consecutive frames. A high frame rate would be required to ensure a high overlap and to minimize the amount of mismatch, as the method is solely based on the size and position of the bounding boxes and not utilizing the image information.

3.5 Stereo matching

Stereo matching becomes trivial after having sorted corresponding features with Bochinski's tracking algorithm. The videos have also been rectified, thus all corresponding points would be located along the same epipolar line. After rectification there should ideally be no disparity along the y-axis and the disparity between the two images can be obtained by subtracting the horizontal position of the corresponding image points. However, computing the disparity from the position without using the image information, assumes that the object detection have a pixel perfect accuracy. Incorrect object detection can result in inaccurate motion estimates. To improve the accuracy a block matching (BM) approach was implemented. The implemented method defines a local region (often referred to as a window) containing the position and its neighbouring pixels in the left image and searches for another window along the epipolar line until it finds the most suitable match in the right image. Different cost functions can be used to measure the similarity between the windows that are being compared, but based on the results from testing, the normalized cross-correlation function (NCC) proved to be the most accurate. The OpenCV implementation of NCC was used as a similarity measure for block matching during the experiments (Bradski, 2000):

$$NCC(x,y) = \frac{\sum_{x',y'}(W(x',y')I(x+x',y+y'))}{\sqrt{\sum_{x',y'} W(x',y')^2 \sum_{x',y'} I(x+x',y+y')^2}}$$

Here "W" is the window containing the feature position and its surrounding pixels that are being evaluated and "I" is the reference image that the window is being compared against. The window with the maximum value would be the optimal match.

3.6 3D reconstruction

Given the camera parameters and a pair of matched features in the two image planes, the coordinates of the 3D point can be obtained by solving the triangulation problem. The 3D point $\mathcal{X}$ would be located at the intersection point of the projection rays from the two cameras centers $\mathcal{C}$ and $\mathcal{C}'$. Hence the location of the 3D scene point can be derived by the geometric relations between the baseline $\mathcal{B}$, focal length f, depth $\mathcal{Z}$ and the horizontal disparity $x - x'$ of the image plane (Figure 4). The final trajectory can then be estimated by computing the 3D scene point for each corresponding features in consecutive frames.

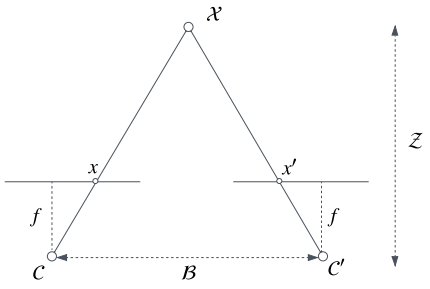

Fig. 4. Geometric relations: 3D coordinates are derived directly from the triangle, which is formed between the two camera centers and the 3D point.

3.7 Sub-pixel accuracy

Sub-pixel estimation is an image registration technique used to obtain a higher accuracy than the pixel accuracy in digital images. Sub-pixel estimation is typically applied as a refinement step after the initial process of finding discrete correspondences (Szeliski, 2022). In an attempt to improve the accuracy after block matching, the gradient cross-correlation (GCC) method of Argyriou and Vlachos (2003) was implemented as a sub-pixel refinement. T he least squares method can be used to fit a quadratic function to the cross-correlation function of the image gradient. The peak of the fitted function would then represent the image disparity with sub-pixel accuracy. Additionally, as convolution becomes multiplication in the frequency domain, the fast Fourier transform (FFT) can be utilized to reduce the computational time.

3.8 Post processing

A Savitzky and Golay (1964) filter was applied to smooth out the estimated trajectory. The Savitzky–Golay (SG) filter uses convolution and polynomial fitting (e.g. least squares) to reduce the amount of fluctuations while keeping the signal tendency of the trajectory intact. Smoothing out the fluctuations while highlighting the overall trend of the trajectory, can simplify speed and length estimations where discontinuities can easily lead to incorrect estimates.

3.9 Analyze speed, compute length of trajectory

Having estimated a trajectory for the fish movement, some additional indicators that can be utilized in behavioural analysis can be extracted: The length of the trajectory can be used as a measure of activity over time (i.e how much does the fish swim) as well as the speed both as an average for the trajectory and as the gradient (i.e. instantaneous velocity) of the trajectory. The length and average speeds will give insights into the overall activity of the fish and can be calculated by the Euclidean distance between the 3D coordinates from one frame, i, to the next:

$$\Sigma_{i=1}^{N} d = \Sigma_{i=1}^{N} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2}$$

When the sum of distances between frames is divided by the amount of frames, it will give speed in units distance per frame. When this is multiplied by frames per second, it will give a true metric speed. For the continuous speed estimates at any given point, the numerical gradient of the distance can be calculated. For all of these calculations it is important that the trajectories are not corrupted

with noise, as can be seen in Figure 5. Without any post processing or sub-pixel refinement, there are jumps from frame to frame of such a magnitude that it obscures the real movement of the fish.

4. SIMULATION AND EXPERIMENTAL SETUP

In this section, the results from the conduced experiments using the real video footage (Section 4.1) and the synthetic data generated in Blender (Section 4.2) are presented.

4.1 Experiments

In the summer of 2021, an experiment aiming to observe the sound response of fish to various stress factors was carried out at the NINA (The Norwegian Institute for Nature Research) research facility. In this work the stereo video data, specifically videos from 9AM and 10AM, is used to test our proposed tracking methodology for data mining fish behaviour information. The data set contains video of fish every 15 minutes, with each video lasting 5 minutes throughout the day of the experiment with natural lighting and associated issues.

4.2 Synthetic data with ground truth

In order to objectively evaluate the used approach for trajectory measurements we created a simulated experiment with the open source graphics software Blender (Community, 2018) which allowed us to define a camera and a scene where a simplified salmon can move on a predefined ground truth trajectory. For our experiment we placed a salmon image as texture on a flat rectangular surface and moved the surface along the trajectory, with a parallel orientation to the camera. The trajectory represented a motion along a circle with radius 0.5 m centered at 2 m in front of the camera (i,e, x,y,z = 0,0,2). A full round along the circle was performed after 100 frames and the exact trajectory of the salmon-eye was determined considering also the actual offset of the eye on the rectangular surface. Intrinsic camera parameters were set to closely represent the stereo-camera used in the real-world experiment. The horizontal field of view was set to 50 degrees and the baseline between the two parallel oriented stereo cameras was set to 0.15m. The principle point is at the center of the created camera images with size 1280×818 pixels.

5. RESULTS

The estimated trajectories using the methodology laid out in this paper when applied to the experimental data (Section 4.1), are shown in Figure 5 and 6. Both figures shows the estimated trajectories where the disparity is computed with and without block matching in addition to one that is smoothed out with a Savitzky–Golay filter. The results presented in Figure 8 and 7 were estimated using the synthetic data generated in Blender (Section 4.2). In Section 3.9 the method for calculating the length of a trajectory is given, when combined with the knowledge of the duration of the trajectories, the speed can be calculated. For validation of the methodology, these calculations are carried out on trajectories estimated from the synthetic data and compared with its ground truth. The results of this comparison, without block matching, with block matching and using block matching with gradient cross-correlation can be seen in Table 1. A plot of the speed throughout the trajectory (calculated with block matching and gradient cross-correlation), is shown in Figure 8.

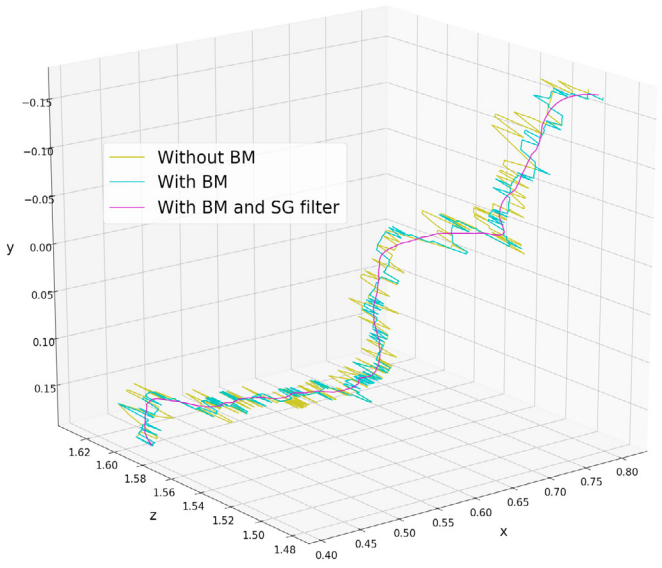

Fig. 5. Trajectory from 9am video recording showing the accuracy differences between a motion trajectory without block matching (BM), with BM, and with BM and a Savitzky–Golay filter.

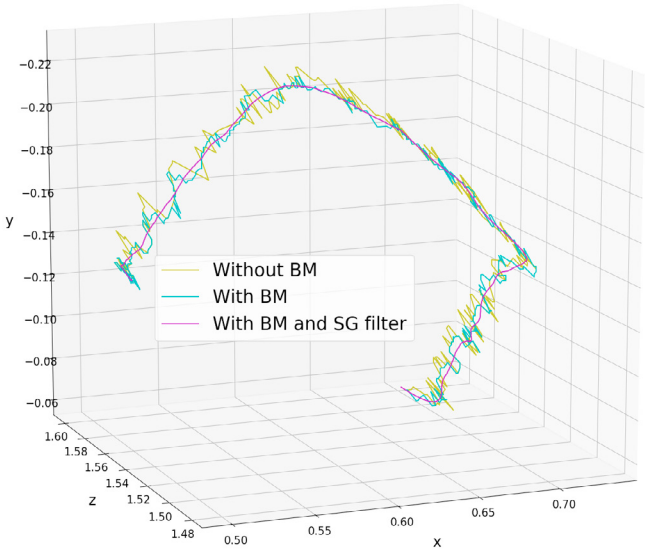

Fig. 6. Trajectory from 10am video recording showing the accuracy differences between a motion trajectory without block matching (BM), with BM, and with BM and a Savitzky–Golay filter.

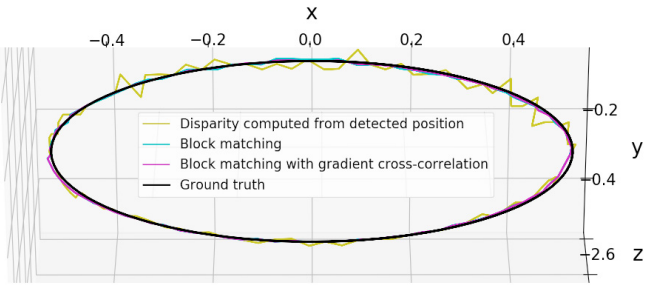

Fig. 7. Comparing the synthetically generated ground truth with the estimated trajectories.

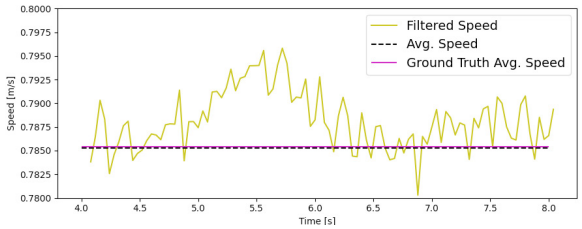

Fig. 8. Filtered instantaneous speed computed from the estimated trajectories obtained with the synthetic data.

Method	Trajectory Length [m]	Speed [m/s]
Ground Truth	7.854	0.7854
Without BM	11.54	1.154
With BM	8.003	0.8003
BM + GCC.	7.853	0.7853

Table 1. Estimated trajectories without and with block matching (BM), with BM and gradient cross-correlation (GCC) as well as the corresponding synthetic ground truth.

5.1 Error measures

In order to evaluate the performance, the estimated trajectory and the ground truth were compared with different error measures. The root-mean-square error (RMSE) emphasize large and undesirable errors in the estimated trajectory. On the other hand, the mean-square error (MAE) is a linear error measure that emphasizes all errors equally. RMSE and MAE were also selected as the preferred error measures as the result would have the same unit as the input data itself. Table 2 show the RMSE and MAE of the euclidean distances that were computed when comparing the estimated trajectory with the ground truth.

Method	RMSE [m]	MAE [m]
Without BM	0.0364	0.0278
With BM	0.009	0.0084
BM + GCC	0.0068	0.0064

Table 2. Resulting RMSE and MAE when compared against the synthetic ground truth of trajectories estimated without block matching (BM), with BM and with BM and gradient cross-correlation (GCC).

6. DISCUSSION

Eye detections of the fish do fail in several cases, primarily due to poor lighting conditions (either over or underexposed regions of the images) as well as the limited field of view causing the fish to mostly be out of the field of view. This prevents tracking over long periods of time. Other issues that adversely affects detections are back scattering and turbidity (Figure 9).

Data augmentation was also found empirically to increase the detection accuracy, this could be due to the artifacts, varying conditions and other data quality issues. The distortion introduced by stereo-rectification, was probably also a factor for why augmentation was needed, as a significant drop in accuracy from non-rectified to rectified images was observed prior to the addition of augmentation.

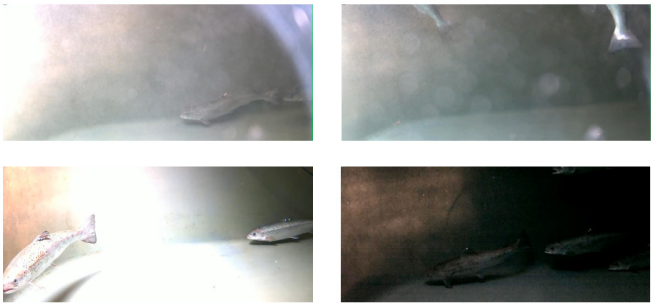

Fig. 9. Challenges: High back-scattering/high turbidity (top left), fish are not visible (top right), overexposed (bottom left), and underexposed (bottom right).

6.1 Stereo matching accuracy

The estimated trajectory from the real experiment shown in Figure 5 and 6, were verified through qualitative comparison with the actual motion of the salmon in the video. Looking at the two figures, one can notice the discontinuous noise along the z-axis. The noise is related to the disparity between the corresponding points that are used in the triangulation process. The noise can be a result of incorrect detections or the images being limited to pixel accuracy, introducing discontinuities in the signal. Sub-pixel refinement can be a viable solution to improve the accuracy. In an attempt to improve the accuracy, a sub-pixel refinement method based on gradient cross-correlation was implemented to estimate the sub-pixel accuracy. A synthetic ground truth was created to quantitatively test the accuracy of each matching method. In Figure 7 one can notice that estimating the disparity directly from the detected feature position leads to the most inaccurate trajectory and that the noise increases when the depth increases. Table 2 shows that there is a difference between estimated trajectories with and without block matching of approximately 2.8 cm. Block matching with sub-pixel refinement yielded the smallest error, but as the same texture of the salmon was used for both left and right image in the synthetic camera setup, further testing is required to evaluate the improvement of sub-pixel refinement. The overall performance can change depending on the image quality and lighting conditions in the testing environment. However, looking at the root-mean-square error in Table 2, it's clear that block matching both with and without sub-pixel refinement can improve the accuracy of the estimated trajectory.

7. CONCLUSION

The definition of welfare indicators is an ongoing research question, especially in the aquaculture industry. With this paper we aim to establish a first step towards this goal, by extracting motion trajectories of individual salmon and swimming speed. With the developed methodology we are able to track the fish while it is within the field of view to retrieve 3D position, motion trajectory and swimming speed. This should provide a good starting point for further work within automatic analysis of behaviour as a

welfare indicator, behaviour remains one of the most influential indicators of animal welfare (Noble et al. (2018)) but also a very challenging indicator to objectively measure. We are aware of that the interpretation of the retrieved analysis results will require an interdisciplinary approach, as similar to other application areas (Stahl et al., 2012). In this work we have taken one step towards objective measurements but further work is needed, in particular testing on full scale farm data. Additional avenues to pursue include better tracking of individuals (such as Romero-Ferrero et al. (2019)) for larger numbers of fish and longer trajectories, further studies into improving the matching methods with even higher sub-pixel accuracy as well as combining multiple features in the tracking, such as head, dorsal and caudal fins, which could improve accuracy as well as provide size and orientation estimates.

REFERENCES

Argyriou, V. and Vlachos, T. (2003). Sub-pixel motion estimation using gradient cross-correlation. In *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, volume 2, 215–218.

Bochinski, E., Eiselein, V., and Sikora, T. (2017). High-speed tracking-by-detection without using image information. In *14th IEEE Inter. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6.

Bochkovskiy, A., Wang, C.Y., and Liao, H.Y.M. (2020). Yolov4: Optimal speed and accuracy of object detection.

Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Chuang, M.C., Hwang, J.N., Williams, K., and Towler, R. (2015). Tracking live fish from low-contrast and low-frame-rate stereo videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 25, 167–179.

Community, B.O. (2018). *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam.

Hartley, R. and Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge Univ. Press.

Huang, T.W., Hwang, J.N., Romain, S., and Wallace, F. (2019). Fish tracking and segmentation from stereo videos on the wild sea surface for electronic monitoring of rail fishing. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(10), 3146–3158.

Jocher, G. (2021). ultralytics/yolov5: v6.0 - yolov5n 'nano' models, roboflow integration, tensorflow export, opencv, dnn support.

Jovanović, V., Risojević, V., Babić, Z., Svendsen, E., and Stahl, A. (2016). Splash detection in surveillance videos of offshore fish production plants. In *Inter. Conf. on Systems, Signals and Image Processing (IWSSIP)*, 1–4.

Li, J., Xu, C., Jiang, L., Xiao, Y., Deng, L., and Han, Z. (2020). Detection and analysis of behavior trajectory for sea cucumbers based on deep learning. *IEEE Access*, 8, 18832–18840.

Lien, A.M., Schellewald, C., Stahl, A., Frank, K., Skøien, K.R., and Tjølsen, J.I. (2019). Determining spatial feed distribution in sea cage aquaculture using an aerial camera platform. *Aquacultural Engineering*, 87, 102018.

Madshaven, A., Schellewald, C., and Stahl, A. (2022). Hole detection in aquaculture net cages from video footage. In W. Osten, D. Nikolaev, and J. Zhou (eds.), *Fourteenth International Conference on Machine Vision (ICMV 2021)*, volume 12084, 258 – 267. International Society for Optics and Photonics, SPIE.

Newbury, P.F., Culverhouse, P.F., and Pilgrim, D.A. (1995). Automatic fish population counting by artificial neural network. *Aquaculture*, 133(1), 45–55.

Noble, C. amd Gismervik, K., Iversen, M.H., Kolarevic, J., Nilsson, J., Stien, L.H., and Turnbull, J.F. (2018). Welfare indicators for farmed atlantic salmon: tools for assessing fish welfare.

Romero-Ferrero, F., Bergomi, M.G., Hinz, R.C., Heras, F.J.H., and de Polavieja, G.G. (2019). idtracker.ai: tracking all individuals in small or large collectives of unmarked animals. *Nature Methods*, 16(2), 179–182.

Savitzky, A. and Golay, M.J.E. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8), 1627–1639.

Sekachev, B., Manovich, N., Zhiltsov, M., Zhavoronkov, A., Kalinin, D., Hoff, B., TOsmanov, Kruchinin, D., Zankevich, A., DmitriySidnev, Markelov, M., Johannes222, Chenuet, M., a andre, telenachos, Melnikov, A., Kim, J., Ilouz, L., Glazov, N., Priya4607, Tehrani, R., Jeong, S., Skubriev, V., Yonekura, S., vugia truong, zliang7, lizhming, and Truong, T. (2020). opencv/cvat: v1.1.0.

Stahl, A., Schellewald, C., Stavdahl, Ø., Aamo, O.M., Adde, L., and Kirkerod, H. (2012). An optical flow-based method to predict infantile cerebral palsy. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(4), 605–614.

Szeliski, R. (2022). *Computer Vision. Algorithms and Applications*. Springer, 2nd edition.

Torisawa, S., Kadota, M., Komeyama, K., Suzuki, K., and Takagi, T. (2011). A digital stereo-video camera system for three-dimensional monitoring of free-swimming pacific bluefin tuna, thunnus orientalis , cultured in a net cage. *Aquatic Living Resources*, 24, 107–112.

Xu, W., Zhu, Z., Ge, F., Han, Z., and Li, J. (2020). Analysis of behavior trajectory based on deep learning in ammonia environment for fish. *Sensors*, 20(16), 4425.

Yang, X., Zhang, S., Liu, J., Gao, Q., Dong, S., and Zhou, C. (2021). Deep learning for smart fish farming: applications, opportunities and challenges. *Reviews in Aquaculture*, 13(1), 66–90.

Zhang, H., Cisse, M., Dauphin, Y.N., and Lopez-Paz, D. (2018). mixup: Beyond empirical risk minimization.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 1330–1334.