

FUSED SURF-FAST Based Particle Filter Tracking of Arapaima Gigas

Faisal Sani Bala¹, Amir Akramin Shafie², Umar Usman Bello³, Zaw Zaw Htike⁴

¹Lecturer, Dept. of Mechatronics and Systems Engineering, Abubakar Tafawa Balewa University Bauchi, Nigeria

²Professor, Dept. of Mechatronics Engineering, International Islamic University of Malaysia, Gombak Malaysia

³Senior System Engineer, Directorate of ICT, Abubakar Tafawa Balewa University, Bauchi State Nigeria

⁴Asso. Professor, Dept. of Mechatronics Engineering, International Islamic University of Malaysia, Gombak Malaysia

Abstract - Invasive Alien Species (IAS) of fish has recently become issue of concern, due to their adverse ecological effect, as well as potential risk and danger to humans. Method of containment mostly employed to invasive fish without harming indigenous fish species would involve direct human effort. The involvement of humans in physical and direct containment of invasive fish species can be very tedious, as it involves diving and hunting of alien fish species. However, the use of vision based underwater robots can greatly reduce the cost, effort and risk involved, as well as yield more result in shorter time. Underwater robot vision system is primarily built upon visual recognition and tracking. Due to the nature of underwater environment, as well as tracking target, it becomes necessary that the underwater tracker should have good performance. In this study, the particle filter tracking algorithm is employed for underwater tracking of Arapaima Gigas, where modifications for its improvement were proposed. The improvement is towards enhancing the tracker performance in terms of accuracy and tracking error using SURF-FAST fused feature for particle filter tracking. The features used for tracking are Speeded Up Robust Feature (SURF) and Fast Accelerated Segment Test (FAST). The result from fused SURF-FAST tracker was the better than single feature FAST or SURF trackers in terms of accuracy and tracking error. However, better performance can be achieved with more training data for Arapaima Gigas.

Key Words: *Arapaima Gigas ;underwater; Robotic vision; accuracy*

1. INTRODUCTION

Invasive Alien Species (IAS) are non-indigenous species whose spread or introduction threatens biological diversity[1]. IAS consist of plants, fish, weeds and other forms of living. The introduction of

these species are mainly due to human activities, it is reported that at least half of the world's invasive species of fish are present in Malaysian waters [1]. Some of the IAS in Malaysia include Pacu (*Piaractus Brachyomus*), Suckermouth Catfish (*Loricariidae Hypostomus Plecomus*), *Collosoma Macroponum* and Arapaima Gigas. Among these invasive fish species, Arapaima Gigas and *Collosoma Macroponum* have been prohibited to be cultured or reared in Malaysian waters, [1]. Arapaima Gigas is the invasive fish species considered in this work, which is native to the Amazon in South America [2].

The fact that Arapaima Gigas is a tropical fish led to its successful survival in the Malaysian ecosystem. However, there is much concern about this specie of fish, due to its big size, behaviour and carnivorous feeding. Aside from human activities such as pollution and over-fishing, rapid decline of indigenous species in Malaysian inland waters have been attributed to the carnivorous alien fish species, such as Arapaima Gigas [3]. Another major concern regarding Arapaima is the risk of attack to local fishermen as well as tourists, which could be a primary cause of death [4].

Currently there are a number of measures adopted by the Malaysian Government to tackle the growth of IAS fish population [1], it includes prevention, containment and eradication. However, more attention is given to preventive measures, such as ban on the importation of invasive alien fish species. Containment and eradication on the other hand have not been implemented due to the high cost and laborious effort associated with them. Thus, invasive alien fishes already in Malaysian ecosystem will continue to thrive, since no eradication measure in place. In similar situation, ocean divers were engaged in the control of crown of thorn fish using bile guns [5].

An alternative method for containment will involve the use of underwater robots, to effectively sterilize or decapitate the invasive fish species. This approach was proposed to control Crown of Thorn Fish in the Australian Coral Reef [6]. A similar approach was proposed for the control of Lion Fish invasion in eastern seaboard of the United States[7]. The underwater robot approach relies heavily on visual tracking.

In this work, an improved particle filter-based method is proposed, using combination of Feature from Accelerated Segment Test (FAST) and Speeded Up Robust Features (SURF). This will form the basis of the automated containment strategy necessary for use with underwater robots.

2. RELATED WORK

Research on computer vision and machine learning application in the study of underwater living things and fishery has developed interest of researchers in recent times. The two major application areas are tracking and classification, the scope of this research is limited to tracking. Abnormality detection and analysis of fish behavior was automated by [8], in the experiment, fish behavior is analyzed in both clean water and chemical contaminated water, through extracting frequent swimming patterns using spectral clustering and trajectory tracing.

A similar work was reported by [9], where covariance based method of fish tracking was used to generate tracks of fish, while a hierarchical classifier based on clustering was used to detect abnormalities in a real-life underwater environment, coupled with erratic movement and low-quality video.

Detection and tracking of multiple swimming fishes in a shallow water with persistent occlusion was reported in [10]. An ellipse fitting and extrema detection technique was employed to detect the fish head, while Kalman filtering alongside a region-based feature matching was used. Trajectory linking was used to cater for occlusion.

A detection and tracking algorithm to be used by under-water robot for administering lethal vial to crown-of-thorn star fish, as an autonomous robotic solution to prevent the fish from destroying coral reef was reported in [6]. The algorithm consisted of a combination of particle filter with random forest

classifier, where probability of the predicted class was used as the measurement for updating the particle filter, while prediction was achieved using optical flow for particle propagation, the detection and tracking were successful in a realistic robotic test bench.

A fish recognition method was presented in [11], based on Generalized Color Fourier Descriptor (GCFD). Features of fish are extracted using GCFD technique, non-relevant information are removed from the spectrum, leaving only the spectrum part containing the fish, the technique was reported to be translation and rotation invariant and has better performance compared with other color descriptors, the method is said to be successful in tracking.

An automatic detection and tracking of fish from video sequences, using Gaussian Mixture Model for identification and Kalman filtering for tracking was demonstrated in [12]. A multi-fish tracking method was proposed in [13] for collective (school) behavior analysis. The authors considered a densely populated fish school, where gliding and bending were analyzed through good tracking information, in relation to energy utilization.

3. PREDICTION

The general framework for Bayesian estimation is given as:

Consider the joint distribution of

$$p(x_t | y_{1:t}) = \frac{p(y_t | x_t) p(x_t | y_{1:t-1})}{p(y_t | y_{1:t-1})} \quad (1)$$

Equation 1 is the marginal distribution where $p(x_t | y_{1:t-1})$ is the state prior and $p(y_t | x_t)$ is the likelihood of data, but $p(y_t | y_{1:t-1})$ is not explicitly known. Thus $p(y_t | y_{1:t-1})$ can be replaced with the variable β . Thus:

$$p(x_t | y_{1:t}) = \beta p(y_t | x_t) p(x_t | y_{1:t-1}) \quad (2)$$

Equation 2 is known as the update equation. While the prediction equation is solved as the Chapman-Kolmogorov Equation as expressed below:

$$p(x_t | y_{1:t-1}) = \int p(x_t | x_{t-1}) p(x_{t-1} | y_{1:t-1}) dx_{t-1} \quad (3)$$

Equations 2 and 3 are solved recursively, with different methods, thus giving rise to different filtering techniques known such as particle filter and Kalman filter

Most often, point estimate of x_t , ($\hat{x}_{t|t}$) is computed as the posterior mean given as:

$$\hat{x}_{t|t} = E[X_t | (Y_{1:t} = y_{1:t})] = \int x_t p(x_t | y_{1:t}) dx_t \quad (4)$$

3.1. Kalman Filters

Kalman Filter is a Markovian observer that uses stochastic least-square to provide best estimate of state. Using noise corrupted or inaccurate measurement at the time instance t-1, the Kalman filter computes the state estimate \hat{x}_t^- and the covariance matrix P_t^- . Kalman filter can be used to solve the filtering problem, particularly when a linear dynamical model and measurement models with Gaussian processes and measurement noise are involved [14].

The state is expressed in equation 5, where A is the matrix which describes the state evolution, while B is the measurement matrix, equations 7 to 9 are solved recursively.

$$\hat{x}_t^- = A\hat{x}_{t-1}^- + B\hat{u}_{t-1}^- \quad (5)$$

$$P_t^- = AP_{t-1}^- A^T + Q \quad (6)$$

$$K_t = P_t^- H^T (HP_t^- H^T + R)^{-1} \quad (7)$$

$$\hat{x}_t = \hat{x}_t^- + K_t (Z_t - H\hat{x}_t^-) \quad (8)$$

$$P_t = (1 - K_t H) P_t^- \quad (9)$$

3.2. Particle Filters

In particle filtering estimation technique, the target density and sampling density are first identified, namely:

$$p(x_t | y_{1:t}) : \text{target density}$$

$$p(x_t | y_{1:t-1}) : \text{sampling density}$$

From Bayes' theorem, the two densities are related as:

$$p(x_t | y_{1:t}) \propto p(y_t | x_t) p(x_t | y_{1:t-1}) \quad (10)$$

Where $p(y_t | x_t)$ is the likelihood

The importance weight is expressed as

$$w_{t|t-1}^{(i)} \equiv \frac{1}{N} \frac{p(x_{t|t-1}^{(i)} | y_{1:t})}{p(x_{t|t-1}^{(i)} | y_{1:t-1})} \quad \forall i = 1, \dots, N \quad (11)$$

Where:

$\{w_{t|t-1}^{(i)}\}_{i=1}^N$ are the unnormalized importance weights

$\{x_{t|t-1}^{(i)}\}_{i=1}^N$ are the independent identical distributed

(i.i.d) samples from $p(x_t | y_{1:t-1})$

The normalized importance weight is computed as:

$$\tilde{w}_{t|t-1}^{(i)} = \frac{p(y_t | x_{t|t-1}^{(i)})}{\sum_{i=1}^N p(y_t | x_{t|t-1}^{(i)})} \quad \forall i = 1, \dots, N \quad (12)$$

$$= \frac{1}{N} \sum_{i=1}^N x_{t|t-1}^{(i)} \frac{p(y_t | x_{t|t-1}^{(i)})}{\sum_{j=1}^N p(y_t | x_{t|t-1}^{(j)})}$$

(13)

$$= \frac{1}{N} \sum_{i=1}^N x_{t|t-1}^{(i)} \tilde{w}_{t|t-1}^{(i)}$$

(14)

The particle filter algorithm can be used for estimation as shown in the algorithm in Figure 1

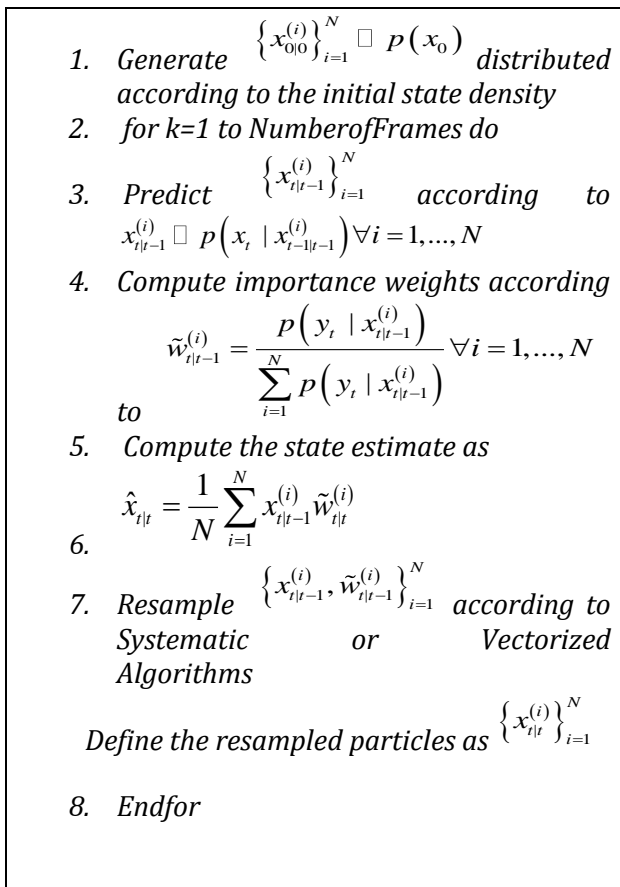


Figure 1: SIR Particle Filter Algorithm

The resampling algorithm of step 6 in the SIR Particle Filter algorithm is realized either using systematic resampling or its fast variant which is vectorized [15] and suitable for MATLAB implementation, the vectorized version is presented in Figure 2.

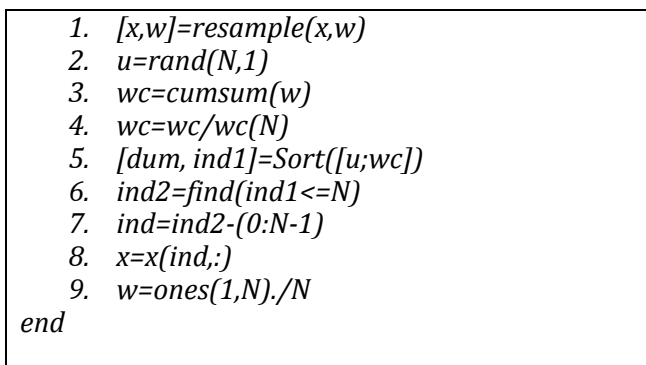


Figure 2: Vectorized Resampling

The motion state of Arapaima Gigas which is assumed to be constant without acceleration is given by a state vector of four dimensions with

displacement and velocity components respectively :

$$\mathbf{x} = [x_t, y_t, v_{x,t}, v_{y,t}]^T \quad 15$$

The typical equations of motion for a target are:

$$\mathbf{x} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_t \\ y_t \\ v_{x,t} \\ v_{y,t} \end{bmatrix} \quad 16$$

To ease of computation, $\Delta t = 1$. The equation $\mathbf{x}_t = F\mathbf{x}_{t-\Delta t}$ can be split into deterministic and stochastic, which helps to account for the noise model. The noise model used is gaussian, given as

$$\mathbf{x}_0^i = \mathbf{x}_0 + N(0, \Sigma) \quad 17$$

The measurement of the Arapaima position is based on the centroid of matched features between the current frame and the template frame. This centroid, which gives the coordinates of x and y respectively in terms of pixels is then compared with the coordinate of everyone of the particles, after which the likelihood is computed

$$(x, y) = [mean(x_{feature}), mean(y_{feature})] \quad 18$$

Where $x_{feature}$ is the vector that contains the x coordinate of the matched features between the template and the current frame and $y_{feature}$ is the y coordinate of the matched features between the template and the current frame.

In order to come up with the centroid of the object, feature matching is performed.

4. FEATURE EXTRACTION

The two classes of features namely the scale invariant point features and corner features are extracted using SURF and FAST algorithms respectively

4.1 SURF Feature Extraction

SURF feature extraction [16] is first applied to template, before it is applied in every frame during tracking. The procedures are outlined below:

1. An integral image was formed using equation 20.

$$\bar{x} = (x, y)^T \tag{19}$$

$$I(x, y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(x, y) \tag{20}$$

2. Using boxfilters, D_{xy} , D_{xx} and D_{yy} are computed, thus yielding the following approximations:

$$L_{xy} \approx D_{xy} \tag{21}$$

$$L_{yy} \approx D_{yy} \tag{22}$$

$$L_{xx} \approx D_{xx} \tag{23}$$

3. The approximate Hessian was computed as:

$$H_{approx} = \begin{bmatrix} D_{xx} & 0.9D_{xy} \\ 0.9D_{xy} & D_{yy} \end{bmatrix} \tag{24}$$

4. A non-maximum suppression is performed on the determinant of Hessian

$$Det(H_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \tag{25}$$

5. Using Taylor's expansion of the Hessian, the extrema was computed:

$$H(\bar{x}) = H + \frac{\partial H^T}{\partial x} + \frac{1}{2} \bar{x}^T \frac{\partial^2 H}{\partial x^2} \bar{x} \tag{26}$$

$$\hat{\bar{x}} = \frac{-\partial^2 H^{-1} \partial H}{\partial \bar{x}^2}$$

27

6. Extract a dominant orientation for each of the interest points
7. Build SURF features from normalized gradient distributions

Using the SURF feature extractor, the points in Figure 3 were detected from Arapaima Gigas tracking.

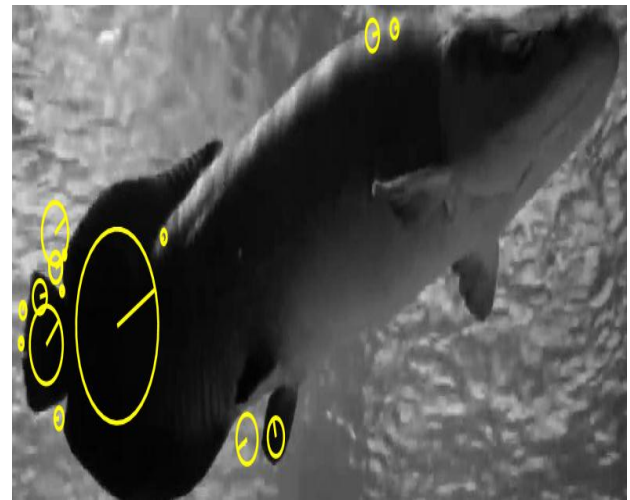


Figure 3: Extracted SURF Features from Arapaima Gigas

4.2 FAST Feature Extraction

The following steps were followed for the FAST feature extraction (Rosten et al., 2010) for both template and tracked object.

1. Choosing candidate pixel for interest point with certain intensity I_p
2. Select some threshold value t .
3. Consider a circle of 16 pixels around the candidate pixel
4. The candidate pixel is a corner if there exists a set of n contiguous pixels in the circle which are either darker than $I_p - t$ or brighter than $I_p + t$ where $n = 12$
5. To exclude large number of non-corners, the test examines four pixels, namely 1,9,5 and 13, starting with 1 and 9, then followed by 5 and 13. P can only be a corner if at least three of the four pixels are either brighter than $I_p + t$, or darker than $I_p - t$, else P is not a corner. Full segment test criterion can then be applied to the passed candidates through examining all the pixels inside the circle.

6. In order to prevent multiple corner detection in the vicinity of the same pixel, the minimum distance between the corners is used, if two corners are very close, only the corner with highest score is retained, the score can be computed as the sum of intensity difference between p and the pixel in the run. The FAST corner feature extraction on Arapaima tracking frame is shown in Fig. 4



Figure 4: Corner Features of Arapaima Detected using FAST

5. PROPOSED METHOD

5.1 Fused SURF-FAST Features

SURF is a scale invariant feature extractors, while FAST is a corner feature extractor. Each of the three feature extractors returns a vector of features, which has coordinates of the feature. Thus, SURFT and FAST are fused, as depicted below:

- Let the feature generated by SURF feature extractor be: F_{SURF}
- Let the feature generated by FAST feature extractor be: F_{FAST}
- Let the feature vector resulting from the SURF and FAST be: $F_{SURF+FAST}$

Using set theory, the fused feature set will be a union of two features, thus:

$$F_{SURF+FAST} = F_{SURF} \cup F_{FAST} \quad 28$$

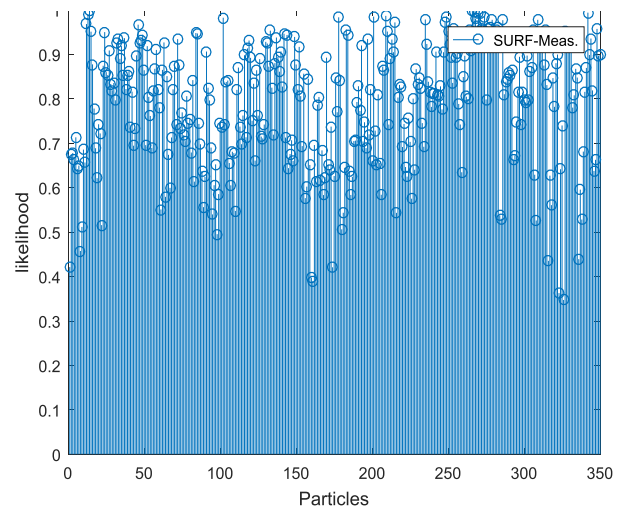


Figure 5: Likelihood Stem Plot of SURF Measurement

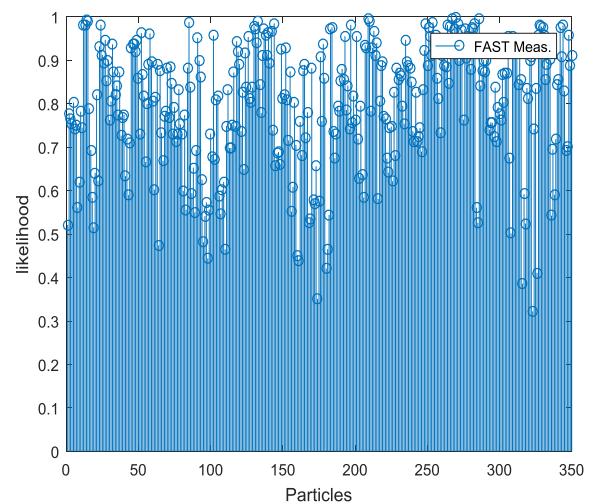


Figure 6: Likelihood Stem Plot of FAST Measurement

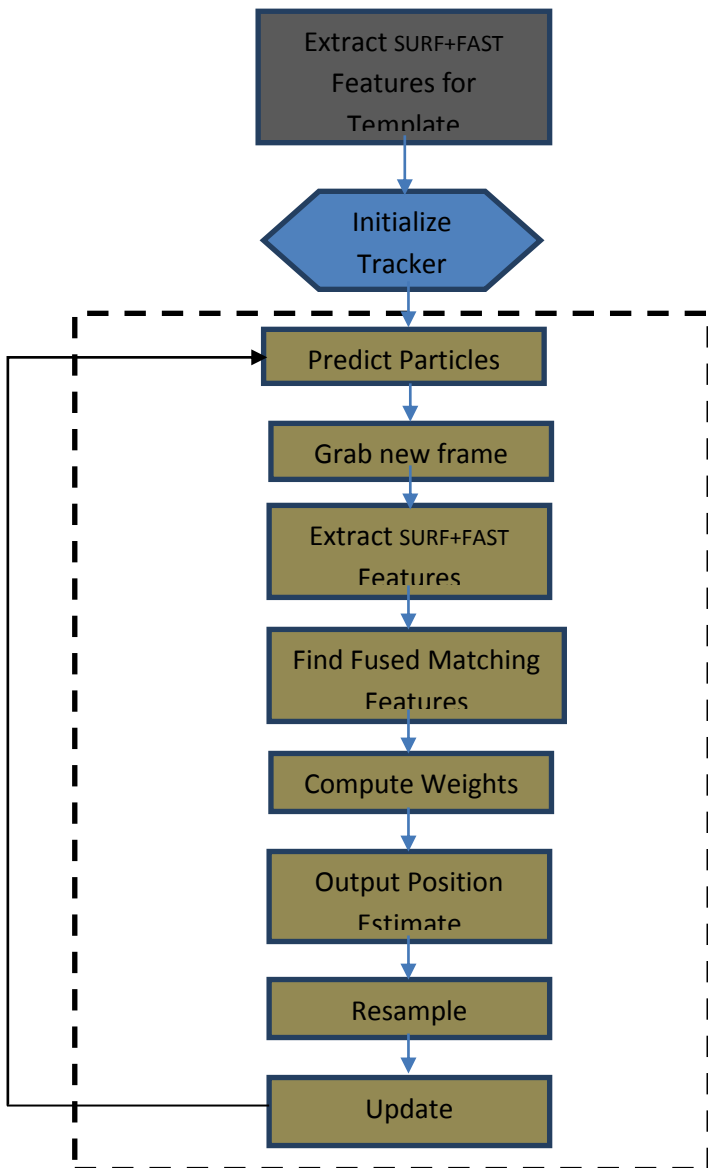


Figure 7: Flow chat of FUSED SURF-FAST Particle Filter

6. RESULT

6.1 Accuracy

In order to arrive at the best of the improved particle filters, it is desirable to employ other measures for evaluation and quantification, such as accuracy which was used by Jiang et. al [17].The expression for accuracy is given in equation 31.

$$a = \begin{cases} 1 & Dev < \sigma \\ 0 & Dev \geq \sigma \end{cases} \quad 31$$

Where

$$\sigma = \frac{\text{length_of_fish}}{10}$$

$$Dev = \| \mathbf{x}_{annot} - \mathbf{x}_{est} \|$$

The length of fish is computed to be 1156, computing the accuracy in percentage, Table 1 is presented

Table 1: Accuracy of Trackers in 500 Frames

| Tracker | Accuracy |
|--------------------|----------|
| PF_SURF | 71.80 |
| PF_FAST | 54.20 |
| PF_SURF-FAST_FUSED | 90 |

6.2 Root Mean Square Error

To further ascertain the best among the trackers, the Root Mean Square Error (RMSE) is employed [18], through this metric, the aggregated error instead of the running error of the trackers can be computed, as well as visualized using a histogram plot. The Root Mean Square Error is computed using the formula in equation 32. The plot is also provided in Figure 9, where the Fused SURF-FAST Particle filter having the least RMSE of 47.5757, SURF Particle Filter having a RMSE of 96.9382 and FAST Particle Filter with the highest RMSE of 112.9458.

$$RMSE = \sqrt{\frac{1}{M} \sum_{i=1}^M (\mathbf{x}_{annot}^i - \mathbf{x}_{est}^i)^2} \quad 32$$

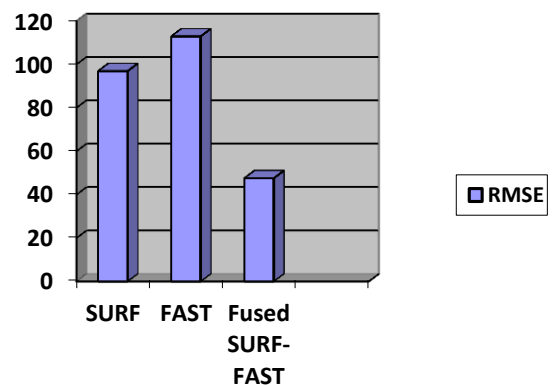


Figure 9: RMSE Plot for the proposed particle filter and single feature particle filters

6.3 Tracking Error

The absolute tracking error was plotted for 500 frames and all the trackers were analyzed. Absolute tracking error is expressed as:

$$e_{abs} = |\mathbf{x}_{est} - \mathbf{x}_{annot}|$$

33

Where

e_{abs} : Absolute tracking error

\mathbf{x}_{est} : Estimated position of fish

\mathbf{x}_{annot} : Ground truth

Plots of the tracking errors are provided in Figures 10 and 11

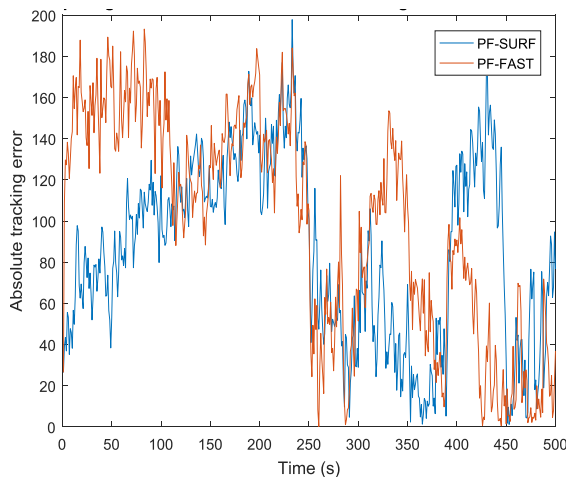


Figure 10: Absolute tracking error of single-feature particle filters

The first plot considered, which is Figure 10 is to evaluate the performance of the individual feature trackers, from the plot of Figure 10, it is evident that the PF_FAST tracker has larger tracking error from the beginning of the tracking as compared to PF_SURF tracker.

Comparing Absolute error Performance SURF, FAST and Fused FAST&SURF PFs

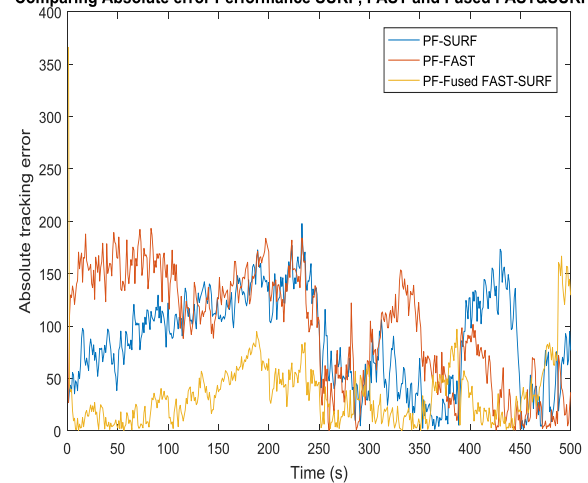


Figure 11: Comparing SURF based and FAST based Particle Filters and the fused SURF_FAST Particle Filter

The particle filter for Fused SURF-FAST feature is compared with the single feature SURF and single Feature FAST particle filters in Figure 11, it is evident that the Fused feature of the two single-feature particle filters outperforms the individual feature particle filters, thus having the least error.

REFERENCES

- [1] Jabatan Perikanan Malaysia., Action plan for aquatic invasive alien species in Malaysia. Marine Fishery Resources Development and Management Department, 2007.
- [2] C. J. Ferraris and C. J. Ferraris, Check List of the Freshwater Fishes of South and Central America. Porto Alegre, Brazil: EDIPUCRS, 2003.
- [3] K. A. A. Rahim, Y. Esa, and A. Arshad, "The Influence of Alien Fish Species on Native Fish Community Structure in Malaysian Waters," Kuroshio Sci., vol. 7, pp. 81–93, 2013.
- [4] R. S. N. Murali, "Giant fish sightings in Kenyir Lake - Community | The Star Online," 2009.
- [5] GBRMPA, "COTS-control-guidelines-update-2017," Gt. Barrier Reef Mar. Park Auth., 2017.
- [6] F. Dayoub, M. Dunbabin, and P. Corke, "Robotic detection and tracking of Crown-of-Thorns starfish," IEEE Int. Conf. Intell. Robot. Syst., vol. 2015–Decem, pp. 1921–1928, 2015.

- [7] E. Anderson and R. Goforth, "Bio-Inspired Robotic Fish With Vision Based Target Tracking," 2014.
- [8] M. Thida, H. Eng, and B. F. Chew, "Automatic Analysis of Fish Behaviors and Abnormality," *Compute*, 2009.
- [9] C. Beyan and R. Fisher, "Detection of Abnormal Fish Trajectories Using a Clustering Based Hierarchical Classifier," *BMVC*, Bristol, UK, pp. 1–11, 2013.
- [10] Z. M. Qian, X. E. Cheng, and Y. Q. Chen, "Automatically Detect and track multiple fish swimming in shallow water with frequent occlusion," *PLoS One*, vol. 9, no. 9, 2014.
- [11] W. P. Lee et al., "Recognition of fish based on generalized color Fourier descriptor," *Proc. 2015 Sci. Inf. Conf. SAI 2015*, pp. 680–686, 2015.
- [12] E. Lantsova, T. Voitiuk, T. Zudilova, and A. Kaarna, "Using low-quality video sequences for fish detection and tracking," *Proc. 2016 SAI Comput. Conf. SAI 2016*, pp. 426–433, 2016.
- [13] K. Terayama, H. Habe, and M. A. Sakagami, "Multiple fish tracking with an NACA airfoil model for collective behavior analysis," *IPSI Trans. Comput. Vis. Appl.*, vol. 8, no. 1, pp. 1–7, 2016.
- [14] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.
- [15] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7 PART 2, pp. 53–81, 2010.
- [16] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3951 LNCS, pp. 404–417, 2006.
- [17] Y. Jiang, J. Fang, Z. Li, J. Yue, Z. Wang, and D. Li, "Automatic tracking of swimming koi using a particle filter with a center-surrounding cue," *Math. Comput. Model.*, vol. 58, no. 3–4, pp. 853–861, 2013.
- [18] Y. Salih and A. Saeed, "Comparison of stochastic filtering methods for 3D tracking," *Pattern Recognit.*, vol. 44, no. 10–11, pp. 2711–2737, 2011.