Research Article
The Use of Vision in a Sustainable Aquaculture Feeding System

Jer-Vui Lee, Joo-Ling Loo, Yea-Dat Chuah, Pek-Yee Tang, Yong-Chai Tan and Wei- Jian Goh
Department of Mechatronics and Biomedical Engineering, Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, Kuala Lumpur, Malaysia

Abstract: Dairy feeding causes significant water pollution. By controlling the proper amount of feed, reducing the waste to minimum will effectively reduce the problem of water contamination. In this project, a Sustainable Aquaculture Feed System (SAFS) has been designed and developed. It can automatically feed the fishes by estimating fishes' appetite through machine vision. The discussion includes design and optimization of the vision system using Labview as well as the integration of various components in the SAFS. With the developed algorithm, the system is able to detect the presence of fishes and count the number of fishes. The outcome is able to estimate and infer the fish appetite. Therefore, the feeding time can be planned ahead. In addition, the system includes a Graphical User Interface (GUI) for monitoring, display the feeding status and sensors reading such as pH, turbidity and temperature.

Keywords: Aquaculture, body, fish count and feeding, machine vision, smart system

INTRODUCTION

The rearing of fish, shellfish and some aquatic plants to supplement the natural supply is an approximate equivalent in fishing to agriculture (Hoiberg and Ramchandani, 2000). Fish are reared under controlled conditions all over the world. Fish can be confined in earth ponds, concrete pools, barricaded coastal waters, or cages suspended in open water. In these situations, the fish can be supplied with adequate food and protected from many natural predators (John, 2011).

One of the biggest problems in aqua culturing or fish farming is the inefficiency of the system to convert feed into fish. Excess of feeds contribute to wastes and from a viewpoint of waste management, waste in fish farming differs significantly compared to farm animal wastes. It produces negative environmental effect. By controlling proper amount of feed can greatly reduce waste. But measurement of actual feed intake is nearly impossible. As a result, all those excess, unconsumed feeds contribute to a huge waste output in most of the operation. On the other hand, containment of all these wastes is very hard, the wastes are easily dispersed back into the surrounding water, hence contaminating the water (Cho and Bureau, 2001).

The use of computer vision technologies in aquaculture: Zion (2012) reviewed and highlighted the application of computer vision in aquaculture has made some progress, but has not matured into a useful tool.

Generally, it is used for counting (Yada and Chen, 1997; Zion et al., 2006; Alver et al., 2007), size measurement and mass estimation (Balaban et al., 2010a, b; Gumus and Balaban, 2010; Hufschmied et al., 2011; Zion et al., 2012), gender detection and quality inspection (Di Marco et al., 2009; Odone et al., 2001; Merz and Merz, 2004; Zion et al., 2008), species and stock identification (Zion et al., 2000; White et al., 2006) and the monitoring of welfare and behaviour (Oppedal et al., 2001; Parsonage and Petrell, 2003; Conte, 2004; Xu et al., 2006; Rodriguez et al., 2011).

Counting: Commercially available fish-counting devices are designed for various ranges of fish sizes. The commercial methods and products include Vaki's ''Bioscanner'', ''Nano'' and ''Macro'' (Vaki Aquaculture Systems Ltd., Iceland), AquaScan's ''Fishcounters'' (AquaScan AS, Norway), SRI's ''Fish Counter'' (Smith-Root Inc., USA), Impex's TPS counters (Impex Agency Hoerning ApS, Denmark), AGM Rognsorterer (Maskon AS, Norway) and ''Larcos'' (Spinnaker Electronics Corp, USA).

Other than the commercial methods, Yada and Chen (1997) used a weighing device to count seedling fry but limited only to a few milligrams. It required analytical scales, a protected environment and is too slow to complete the task. Alver et al. (2007) developed an accurate autonomous rotifer sampling and counting system, which extracts samples of rotifers from first-feeding tanks and takes multiple images of known volumes of sample while back-lit. Zion et al. (2006) has
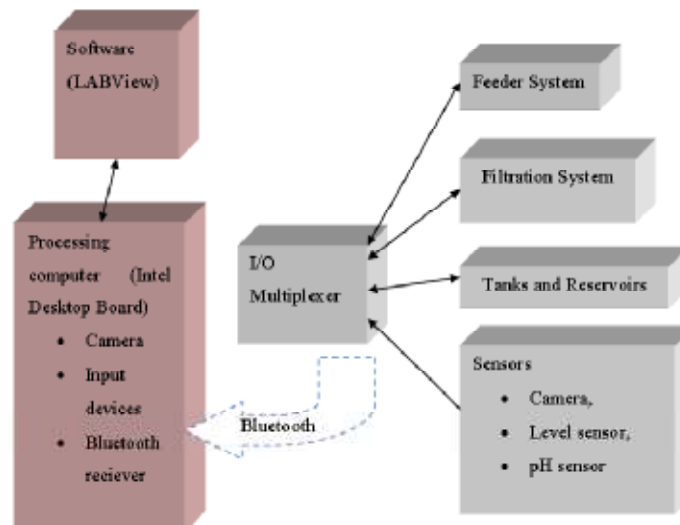
Fig. 1: SAFS architecture

developed a computer vision system based on processing images of batches of fry harvested from the broodstock containers which minimizes the labour cost with the task as well as to reducing the counting errors. Nevertheless, the inspected subjects in aquaculture are sensitive, easily stressed and free to move in an environment in which lighting, visibility and stability are generally not controllable and the sensors must operate underwater or in a wet environment (Zion, 2012).

METHODOLOGY

Sustainable Aquaculture Feeding System (SAFS) architecture: As shown in Fig. 1, the SAFS comprises of a few components, which includes the hardware PC, the software with the database of the fish and the Graphical User Interface (GUI) for controlling and monitoring the fish culturing system. The hardware PC is interconnected with an I/O multiplexer and it acts as an intermediate communicator between all actuators, sensors and the PC. A LABView program is run in Microsoft Windows 7 platform on Intel Desktop Board. And it is interfaced with the I/O multiplexer through the Bluetooth connection.

The information is collected through various sensors. Then the data is processed with preloaded formula before sending to the selected output units. The smart system is able to be operated fully automatically or manually upon end-user's control.

Software: The software as well as the GUI was developed using LabVIEW that is a system design platform and development environment for a visual programming language from National Instruments.

Fig. 2: Intel desktop board D525MW

The developed window-based software enables end-user to access:

- All sites and systems
- Intelligent alarm and event system
- Control of feeding systems
- Full monitor of recirculation system/filtration system
- Full control of cameras
- All sensors reading. (temperature, pH, ammonia, turbidity)

Hardware: The hardware used in developing the SAFS is listed as below:

- Intel Atom Processor D 525
- Intel Desktop Board D 525 MW
- Maxtor 120 Gb harddrive
- 1Gb DDR ram
- Logitech 720 p webcam
- Input devices (mouse and keyboard)

Fig. 3: 720p webcam

Fig. 4: BluetoothBee signal transmitting board

Fig. 5: Sample image of fish in culturing tank
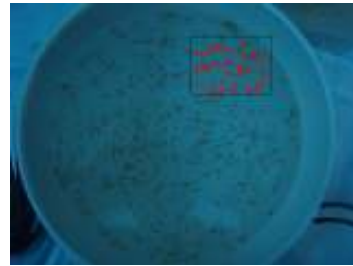

Fig. 6: Sample output image

Fig. 7: Processed sample image after segmentation

- Vztec Bluetooth dongle
- 400 W power supply unit

Figure 2 and 3 show the components of the SAFS. A high definition 720 p webcam is used to provide a and Vztec Bluetooth dongle, which is connected to Intel Desktop Board using USB 2.0. The Bluetooth Bee board is powered by a 9 V battery and able to operate up to 15 m away from the main Intel Desktop Board, which is sufficient in the case of an indoor aquaculture farm. Figure 4 shows the Bluetooth Bee board, including an on-board display for the digital signals received and sent.

Object count: Dairy feeding causes significant water pollution. By controlling the proper amount of feed, reducing the waste to minimum will effectively reduce the problem of water contamination. In achieving the appetite analyzing task, machine vision is applied. By feeding the fish with a small amount of feeds to "test" the appetite of fish within a controlled area, a camera monitors the fish behaviour and the movement at the feed spot. Using image analyzing palette, with feature extraction and vision system, the fish counting task can be achieved.

The palette applied in LabVIEW was Object Count VI 2, which is capable of analyzing different type of image and extracts information out of it. To test the fish appetite, 10 pallets of fish food is feed into the corner of the tank as shown in Fig. 5. Picture is sent to the software for extracting the "contrast" feature.

In this sample fish tank image, the pixels size of 30 was selected for optimal fish counting value. As shown in Fig. 6, some unwanted boundary which has the similar contrast as the actual fish in the tank are also included in the counting process, this can be removed by carefully identifying the "ID" of the selected object and uncheck it from the list. In turn, it will archive a more accurate fish counting results.

Another useful feature of LABView is none other than the "Object Count VI 2", it has a built in segmentation function which ease the user in building the script. As shown in Fig. 7, after the sample image has been segmented, the fish counts at a selected area is achieved, this made the appetite "test" method possible

better clarity of picture or video in the project. A good webcam ensure crisp video quality and avoid unwanted noise due to poor CCD sensor. All the signals from various sensors are managed by I/O multiplexer. Then, the digital signals are sent through Bluetooth connection between two paired devices, Bluetooth Bee

by counting the numbers, or the frequency of the fish at the segmented area.

The number of the fish at normal state is first recorded in this tank. The rough estimated fish counts in the segmented area in normal "non-feeding" states are 20 and it is set as the control value for "normal". During the feeding, the frequency of the fish will increase, with the assumption of the following value:

- 20-25 counts = not hungry (same as control value = average fish count)
- 26-30 counts = a little hungry
- 31-35 counts = quite hungry

- 36-40 counts = hungry
- 41 and above = very hungry

By this method, a rough estimation of fish appetite can be determined before the full load of fish pallets are release into the tank

RESULTS AND DISCUSSIONS

Labview program and its GUI: Labview program as well as the GUI has been design and developed as shown in Fig. 8. It consists of two major parts:
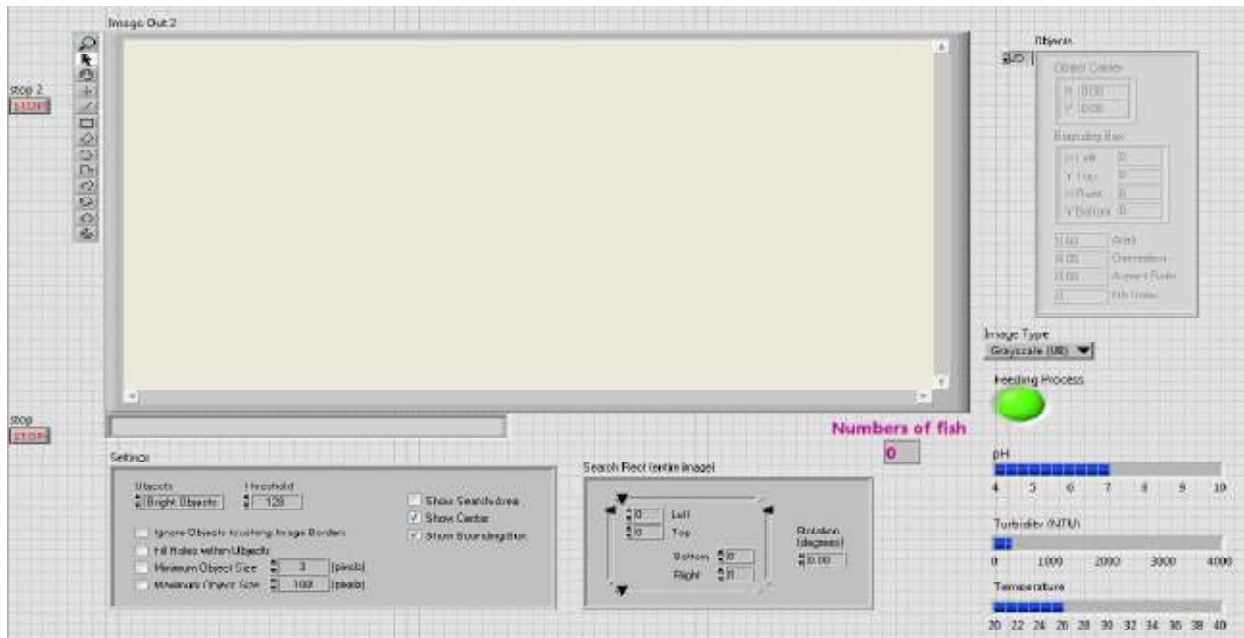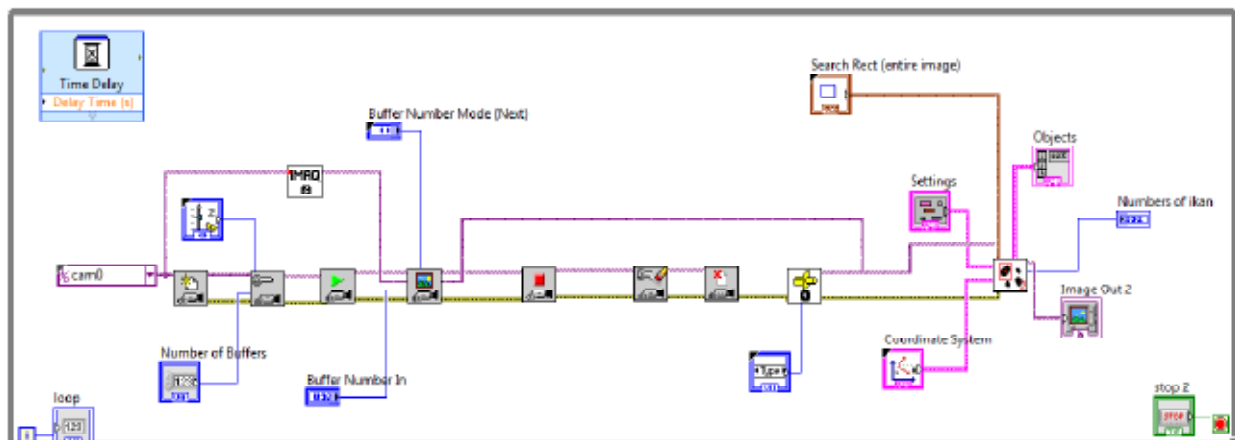

Fig. 8: Main LABView GUI

Fig. 9: The block diagram for object counts in IMAQ Labview

- IMAQ (image acquisition) with object counts
- Time and Date

Figure 8 is the main LABView GUI whereby user can monitor the fish status effectively as well as various. sensors reading which are included in the system. The system will constantly monitor the movement of fish and detect them as an object at the pre-defined section while feeding process. User can also add settings and fine-tune the algorithm to suit different environment. Since the system is very sensitive to lighting and background colour, so whenever a new environment is introduced, all the settings have to be adjusted accordingly to the lighting situation to obtain an optimised result. Figure 9 is the block diagram for image acquisition and the object count template

IMAQ with object counts: The image/video is fed through the webcam and in Labview, this is obtained through the "cam0" setting as shown in Fig. 10. Whereby the images are then acquired and passed through a series of frame grabber setting. The image obtained at grabber is then sent to "object count" for further analysis, at the same time the image type is configured to suit the processing algorithm as the "object count" palette can only process grayscale 8-bit

Fig. 10: Frame grabber block in Labview and image type setting
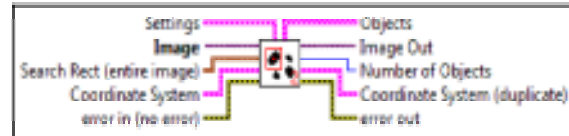

Fig. 11: IMAQ count objects 2 VI Palette

images, so the acquired image has to be converted into grayscale 8-bit before processing. The settings as shown in Fig. 11 is a cluster defining the parameters of the algorithm and the information that is overlaid on the result image. Appendix A summarizes the IMAQ settings. The object count process focuses only a selected area of the image. So the "search Rect" function in this palette is used to section out a rectangular area. This function is summarized in Appendix B.
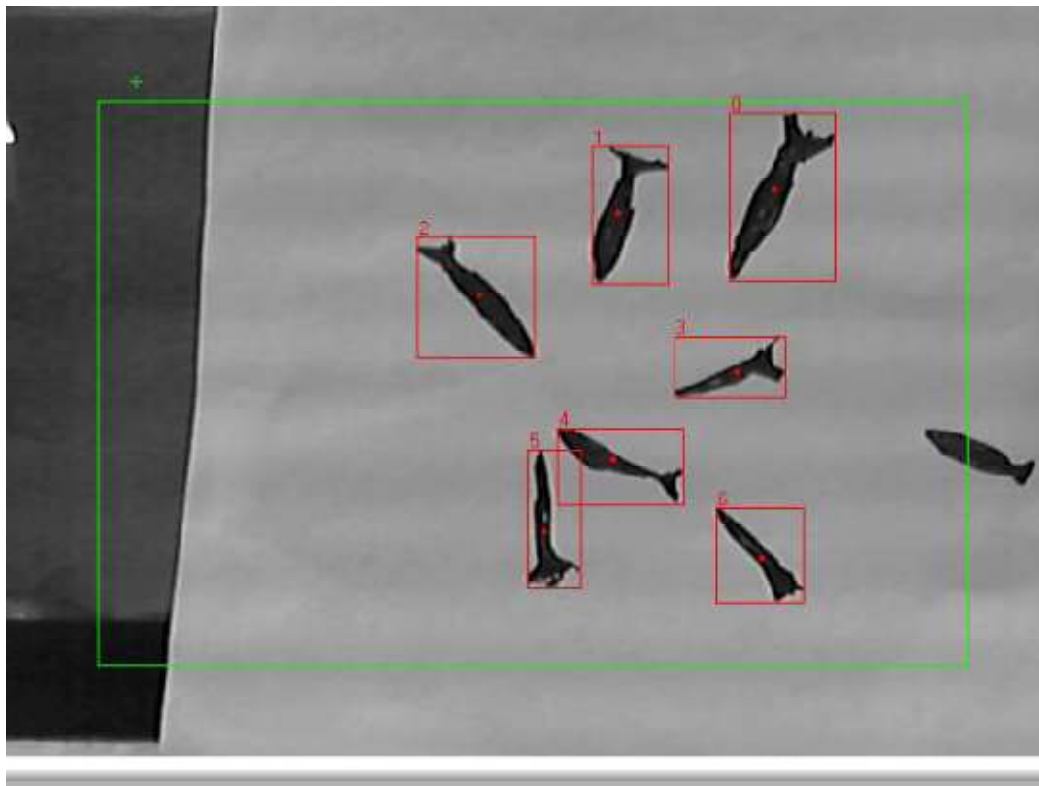

Fig. 12: Successful 7 out of 7 hand-drawn fish detection in Labview

Fish detection with drawn fishes on paper: With the proper area sectioned out for processing, a green rectangle highlights the area to be searched. The result of the hand-drawn fish detection is shown in Fig. 12. The search area and position can be configured by the need of the users. It can be configured at the feed palette drop area and estimate the fish appetite as discussed earlier. The red rectangles are the detection of the fishes, which are labelled from 1 to 7 respectively. The "object count" palette in Labiview successfully detects the intensity difference as well as the position of the intensity changes in the image.

Under the settings as shown in Appendix C, the defining parameters are the object size. It determines the limits of the object to be detected, which can be as

small as 1 pixel, or as big as the entire image. In this project, with the camera placed at a distance of 1meter away from the fish tank, the parameter for the Min and Max size of fish are configured as shown in Fig. 13.

The minimum object size is set to 200 pixels in order to avoid detection of unwanted smaller particles such as the fish waste and the feed palette. Fill holes within objects is to group the fish shinning body part into one fish instead of counting the body and head as 2 separate parts. This reduces the error in fish counting. The maximum object size is to prevent the system from taking the whole dark background shadow as one of the object, hence limiting the number to a maximum size of a fish could avoid this detection problem. However,
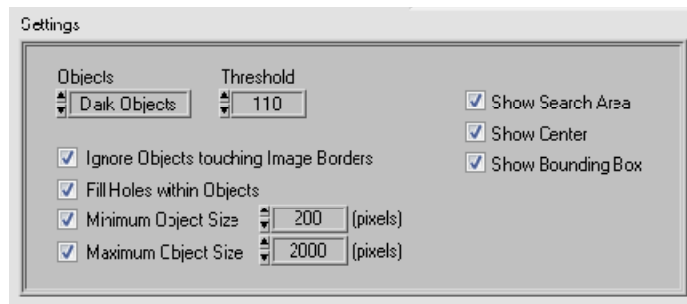

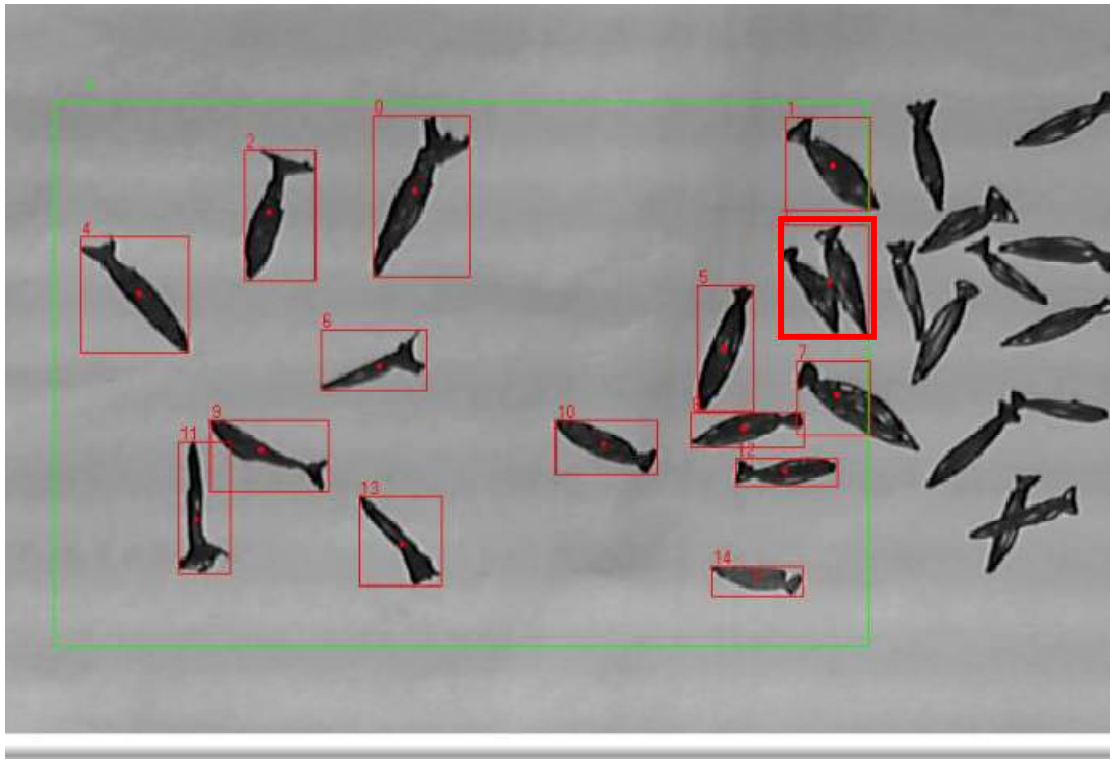Fig. 13: Front panel of object count palette in Labview

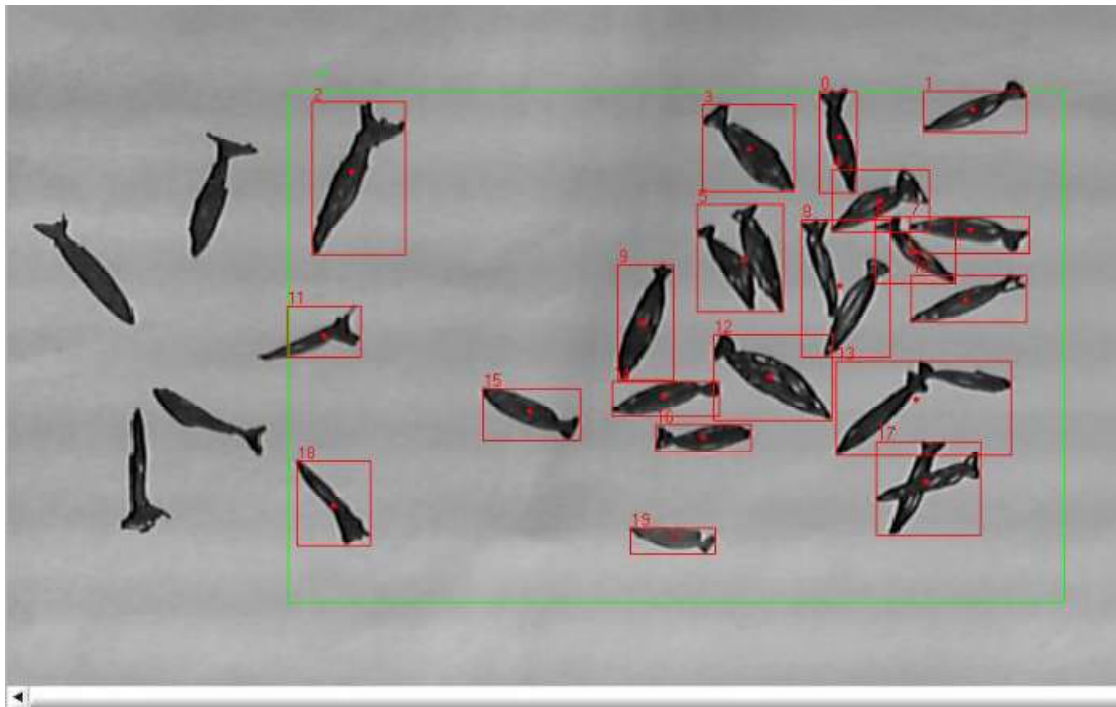Fig. 14: Fish detection when the paper is slide from right to left

Fig. 15: Result of 23 fish detected as 19

if the maximum object size is set too high, 2 distinct fishes might be counted.

When fishes are too closed to each other, faulty detection arises, causing a slight error in the accuracy of detection, but it can be solved by taking the average counts of different images since the real fish swims all the time. Figure 14 is the result when the study is fully slide to the left. The outcome of the detection is acceptable as the accuracy is over 80% using the result obtained from Fig. 15.

Feature extraction depends on the "size" parameter and is very sensitive to it. This parameter is the dominant one in determining a proper object counts. In this case the objects are the fishes, but the problem arises due to the depth of the fish tank. The same size of fish, while swimming higher above the tank looks "bigger" in picture compare with the same fish swimming under. Figure 16 shows the two fishes with same size are viewed differently by the webcam, due to the distance and the perspective of the placement. This can be solved by providing an allowance range of the fish size.

There should be an upper and lower threshold for the software in order to increase the counting accuracy, by introducing a plus minus 50 pixels count. The system will also change the size analysing parameter according to the growth of the fish. Table 1 summarizes the size setting according to the fish growth periodically.
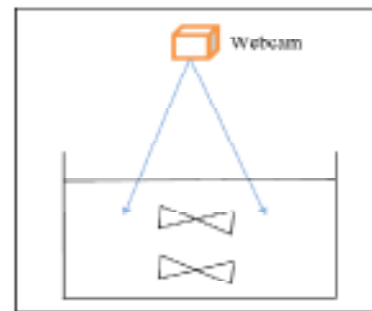

Fig. 16: Different sizes due to the camera perspective

Table 1: Size setting according to the fish growth periodically

Month	Size (pixels count)
3	250 ±50
6	300 ±50
9	350 ±50
12	500 ±50

When the fish grows larger and bigger, the error relative to the size of the fish will be smaller, hence theoretically, this object extraction will be more and more accurate as fish grows bigger. The reason size matter is the proper values of the parameter will successfully "breakdown" larger "merged" objects into separate fish, hence giving more accurate counts as shown in Fig. 17.

Although this solves the problem when 2 fish are very near each other, but it does not solve the problem
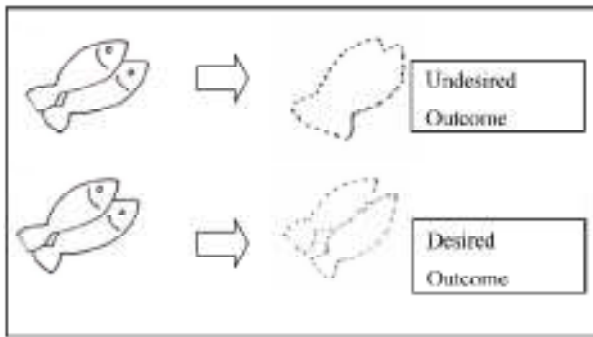
Fig. 17: Correct size successfully "breaks" fish into the correct counts

Fig. 18: Aquarium setup in the lab

if fishes are overlapped on one another. Unfortunately, this is the constraint of top view camera system.

Fish detection of the real fish in the tank: The real fish detection was carried out by mimicking the tank in an aquarium. The background of the aquarium was white in colour that provided a better contrast of fish to the camera. The aquarium setup is shown in Fig. 18. The camera distance set to be 1 m and the images were processed by the LABView program.

The image captured as shown in Fig. 19, the reflection of the water and aquarium itself is quite obvious, whereby to some extend might cause problem to the accuracy of fish count, as the shadows and objects reflected may cause intensity to drop hence detected as a dark object. But since the fish culturing tank will be placed in a controlled environment, the position of lighting is also assumed to be controllable.

Blurred effect problem in real fish detection: There are also a lot of problems in detecting the fishes due to the imperfection of the hardware itself as well as the environment (Fig. 20). Below are some of the few problems encountered in the images.

- Blurry effect due to the frame rate of the web-camera used, causing the blurred fish undetected in the algorithm.
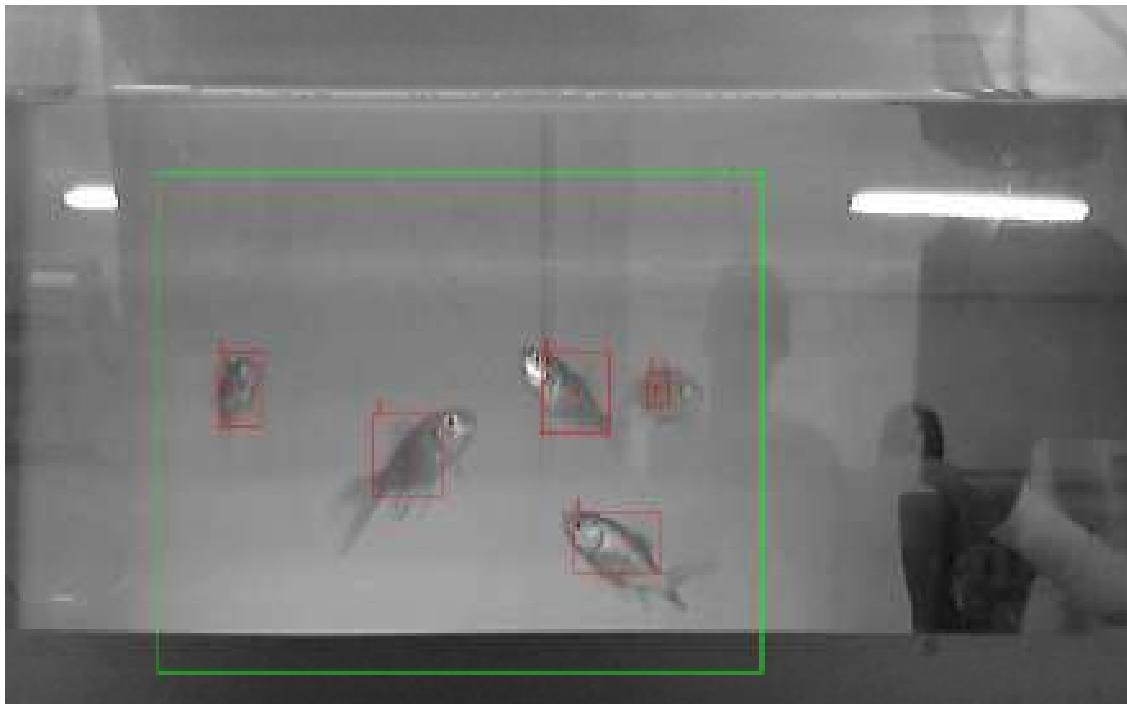- Two fish stacked over each other, hence counted only as one.

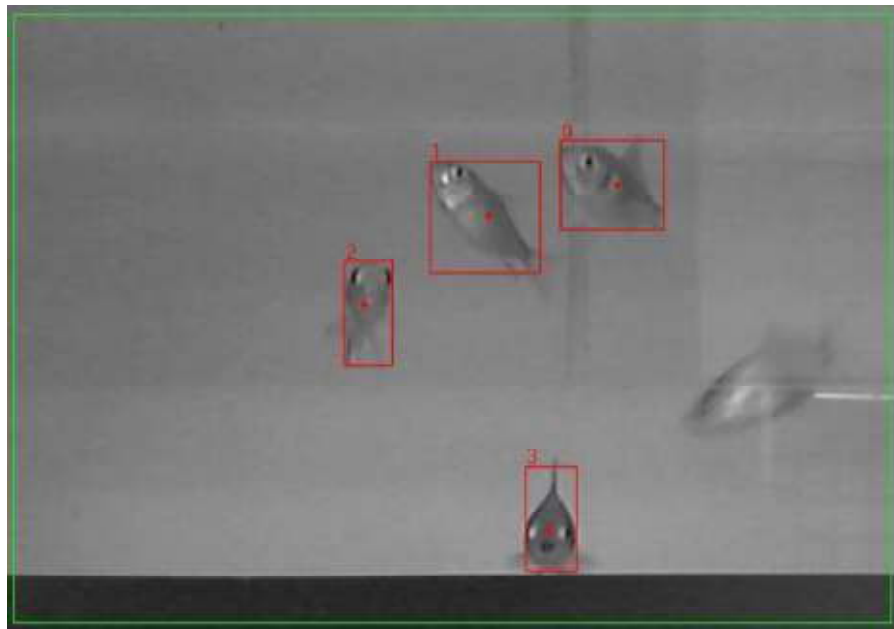Fig. 19: Successful detection of 5 fishes in the aquarium

Fig. 20: Blurred fish on the right goes undetected

Table 2: Gray level profile of the edge with the grey levels shown as heights (for fish at rest and fish swims)

(a)Fish at rest	(b)Fish swim fast
(c)	(d)

Fig. 21: Two fishes detected as one

- Environment light intensity changes, causing background to have darker intensity as well and hence false detection.
- Reflection of fish's skin, causing one fish to be counted as two due to the high brightness of the stomach part.

When the fishes are not moving, the intensity changes of grayscale from background to the fish's body is steeping. Therefore, there is no problem in detecting as shown in Table 2a. But when the fish swims in a relatively fast speed as shown in Table 2b, blur effect occurs due to the low frame rate of the camera. This results in a smooth change in the intensity of grays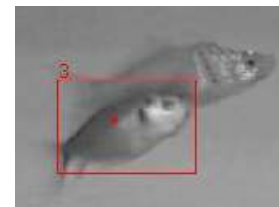cale and therefore it is excluded in object counting. An edge in a gray scale image occurs when there is a transition in gray level over an amount of pixels. A perfect edge would be a transition from black to white over one pixel as shown in the Table 2c. In many images, edges like these won't occur unless it's a binary image. The transition will be blurred spreading the transition over more pixels, resulting in a slope-like profile of the gray level transition as shown in Table 2d. Here the edge is spread over more pixels and will show as a wider edge instead of the 1-pixel transition that shows as a 1-pixel edge. When the transition is that such as Table 2c, it is treated as edge or as an object when analyzed entirely. If the transition is spread too wide, it will be ignore and not treated as object instead.

Multiple fish detected as one in real fish detection:
When two fishes are too closed to each other or sometimes overlapped and the fish behind has slightly higher brightness than the front, the fish behind might end up undetected. The threshold value used is 110 and the result is shown in Fig. 21. The fish behind with slightly low grey intensity is undetected in this
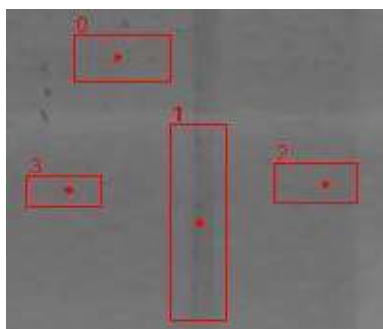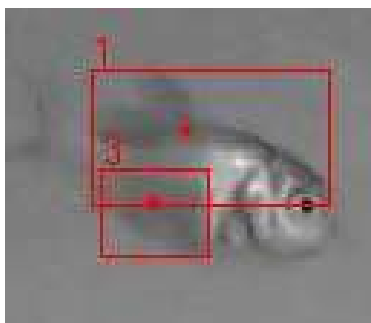
Fig. 22: Unwanted background shadows

Fig. 23: One fish detected as two

threshold value. This is due to the lack of optimization of the "threshold" setting under Object Count palettes. It can be fine tuned to suit one situation at a time. For example, if the threshold is set too high, it will ignore the fish with bright reflection. In the other hand, if it is set too low, it will sometimes take unwanted dark background into account.

Unwanted background noise detected as object: When threshold is set to 60, dark intensity objects are taken into account easily compared to threshold set at 110. If the threshold value is too low, according to this lighting condition, the system will pick up unwanted noise as shown in Fig. 22.

This kind of unwanted count can lead to misinterpretation. The extra counts are interpreted as good appetite in the system and eventually cause the system to overfeed the fish. Eventually, it causes unwanted pollution to the culturing system. So, the algorithm for "threshold" and "object size" must be optimized accordingly to the environment and preferably under a controlled lighting condition. Figure 23 demonstrates another problem encountered when the "threshold" and "object size" are not optimized, which count one fish as two distinct objects. It happens when the threshold is set too low. The reflecting fish's body is considered as a high brightness background region whereas the dark fin on top and bottom are detected separately as 2 distinct objects.

Time and Date: Generating a proper time and date settings is essential in keeping a proper feeding time, in Labview, the time function generator is used to configure the hours per feed as well as long term monitoring in feed quantity as the fish grows. The block diagram is shown in Fig. 24.

An example with 15 sec feeding interval is shown in Fig. 25. The purpose is to test if the signal is sent out to the feeder every 15 sec interval. If signal is received, then the system will adjust the time interval alter the feeding period. When the setting time, for example 4 h is reached, it will send the signal to the feeder system to
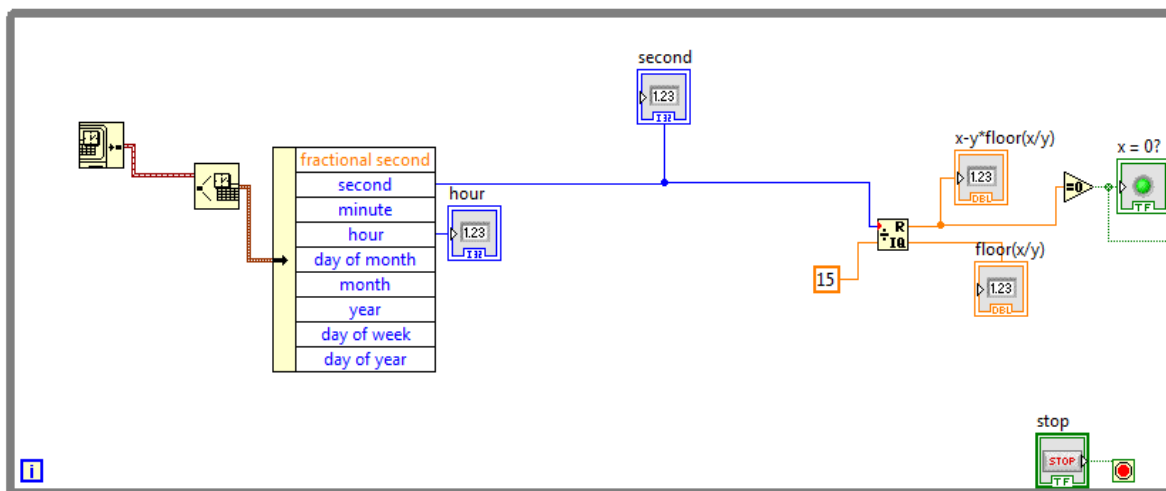

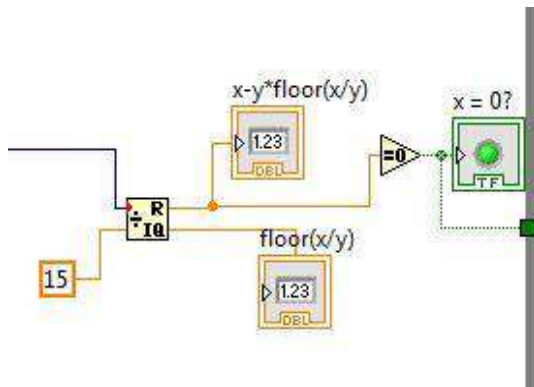Fig. 24: Time generator block diagram

Fig. 25: Boolean for hours between feed

start the feeding process. Each time the feeding cycle starts, the image analysis of fish count will start to estimate the appetite of fish before feeding it with more food palettes.

CONCLUSION

A Sustainable Aquaculture Feeding System (IAFS) has been designed and developed. It includes algorithms that detect and count the fishes and display the sensors signals via a Graphical User Interface (GUI). Below are the summarised of features of IAFS:

- **Fish monitoring:** At 3 frames per second, the video feed of the tank is displayed on main user interface.
- **Fish detection and count:** Successfully detected and counted fishes with 80% accuracy.
- **Sensors reading displays:** pH, Turbidity and temperature are exhibited in bar graphs under the GUI.

Feeding status: User can monitor the feeding process through the main user interface.

APPENDIX

Appendix A

U16	Objects specifies the type of objects to detect Bright objects (0): Searches for dark objects Bright objects (1): Searches for dark objects
U16	Threshold specifies the grayscale intensity that is used as threshold level. When a Bright objects type is selected, the threshold range used includes Threshold to the maximum possible intensity depending of the image type (255 for a 8-bit image). When a Dark Objects type is selected, the threshold range used includes the minimum possible intensity depending of the image type (0 for a 8-bit image) up to Threshold.

Appendix B

EQ6	Search Rect (entire image) specifies the rectangular region of the image where the objects are located. The default value specifies a search on the entire image.
I32	Left is the x-coordinate of the upper left corner of the rectangle
I32	Top is the y-coordinate of the upper left corner of the rectangle.
I32	Right is the x-coordinate of the bottom right corner of the rectangle.
I32	Bottom is the y-coordinate of the bottom right corner of the rectangle.
DBL	Rotation specifies the rotation angle in degrees of the rectangle with its center as point of rotation. If the rotation angle does not equal zero, the Left, Top, right and bottom coordinates are not the actual coordinates of the upper left and bottom right corner of the rectangle, but their position if the rotation angle equals zero.

Appendix C

TF	Fill holes within objects specifies whether the holes in the binary objects are ignored.
TF	Minimum object size specifies whether the detected objects smaller than Min Size are ignored.
I32	Min size specifies the size of the objects to ignore. When the parameter minimum objects size is set to TRUE, the objects which size is smaller than or equal to Min size are ignored.
TF	Maximum object size specifies whether the detected objects larger than Max size are ignored.
I32	Max size specifies the size of the objects to ignore. When the parameter maximum Object Size is set to TRUE, the objects which size is larger than or equal to Max size are ignored.
TF	Show search area determines whether to overlay the ROI on the image.

REFERENCES

Alver, M.O., T. Tennoy, J.A. Alfredsen and G. Oie, 2007. Automatic measurement of rotifer brachionus plicatilis densities in first feeding tanks. Aquacult. Eng., 36: 115-121.

Balaban, M.O., M., Chombeau, D. Cirban and B. Gumus, 2010a. Prediction of the weight of Alaskan pollock using image analysis. J. Food Sci., 75: 552-556.

Balaban, M.O., G.F.U. Sengor, M.G. Soriano and E.G. Ruiz, 2010b. Using image analysis to predict the weight of Alaskan salmon of different species. J. Food Sci., 75: 157-162.

Cho, C.Y. and D.P. Bureau, 2001. A review of diet formulation strategies and feeding systems to reduce excretory and feed wastes in aquaculture. Aquac. Res., 32(S1): 349-360.

Conte, F.S., 2004. Stress and the welfare of cultured fish. Appl. Anim. Behav. Sci., 86: 205-223.

Di Marco, P., V. Donadelli, A. Longobardi, I. Corsalini, S. Duarte, L. Reig and J. Oca, 2009. Measurement of sole activity by digital image analysis. Aquacult. Eng., 41: 22-27.

Gumus, B. and M.O. Balaban, 2010. Prediction of the weight of aquacultured rainbow trout (*Oncorhynchus mykiss*) by image analysis. J. Aquat. Food Prod. Technol., 19: 227-237.

Hoiberg, D. and I. Ramchandani, 2000. Student's Britannica India. Encyclopedia Britannica (India), New Delhi, 7: 29-33.

Hufschmied, P., T. Fankhauser and D. Pugovkin, 2011. Automatic stress-free sorting of sturgeons inside culture tanks using image processing. J. Appl. Ichthyol., 27: 622-626.

John, P.R., 2011. Biomes and Ecosystems. Britannica Educational Publishing, New York, pp: 171.

Merz, J.E. and W.R. Merz, 2004. Morphological features used to identify chinook salmon sex during fish passage. Southwest. Nat., 49: 197-202.

Odone, F., E. Trucco and A. Verri, 2001. A trainable system for grading fish from images. App. Art. Int. 15: 735-745.

Oppedal, F., J.E. Juell, G.L. Taranger and T. Hansen, 2001. Artificial light and season affects vertical distribution and swimming behaviour of post-smolt atlantic salmon in sea cages. J. Fish Biol., 58: 1570-1584.

Parsonage, K.D. and R.J. Petrell, 2003. Accuracy of a machine-vision pellet detection system. Aquacult. Eng., 29: 109-123.

Rodriguez, A., M. Bermudez, J.R. Rabunal, J. Puertas, J. Dorado, L. Pena and L. Balairon, 2011. Optical fish trajectory measurement in fishways through computer vision and artificial neural networks. J. Comput. Civ. Eng., 25: 291-301.

White, D.J., C. Svellingen and N.J.C. Strachan, 2006. Automated measurement of speciesand length of fish by computer vision. Fish. Res., 80: 203-210.

Xu, J.Y., Y. Liu, S.R. Cui and X.W. Miao, 2006. Behavioral responses of tilapia (*Oreochromis niloticus*) to acute fluctuations in dissolved oxygen levels as monitored by computer vision. Aquacult. Eng., 35: 207-217.

Yada, S. and H. Chen, 1997. Weighing type counting system for seedling fry. Nippon Suisan Gakk., 63: 178-183.

Zion, B., 2012. The use of computer vision technologies in aquaculture: A review. Comput. Electro. Agric., 88: 125-132.

Zion, B., A. Shklyar and I. Karplus, 2000. In-vivo fish sorting by computer vision. Aquacult. Eng., 22: 165-179.

Zion, B., N. Doitch, V. Ostrovsky, V. Alchanatis, R. Segev, A. Barki and I. Karplus, 2006. Ornamental Fish Fry Counting by Image Processing. Agricultural Research Organization, Bet Dagan.

Zion, B., V. Alchanatisa, V. Ostrovskya, A. Barkib and I. Karplusb, 2008. Classification of guppies' (Poecilia reticulata) gender by computer vision. Aquacult. Eng., 38(2): 97-104.

Zion, B., V. Ostrovsky, I. Karplus, G. Lidor and A. Barki, 2012. Ornamental Fish Mass Estimation by Image Processing. Agricultural Research Organization, Bet Dagan.