Fish Detection by LBP Cascade Classifier with Optimized Processing Pipeline

DANG HOANG ANH†1 SRIPRASERTSUK PAO†1 KAMEYAMA WATARU†1

In this paper, we present a fish detection and classification mechanism using LBP feature. In this system, training sets are created for each fish species. Besides the strict sample alignment, a processing pipeline is applied in both training and detection process to achieve high performance of detection task. This pipeline further highlights unique features of each species such as edges and dominant colors. Machine learning is used to find the best pipeline model to be applied for each training set. A case study at a local aquarium shows high accuracy at a compelling detection rate.

1. Introduction

Fish detection and classification based on visual information promises valuable applications in many fields including biology, management or even entertainment. Even though the study of fish discrimination is well conducted for more than a decade, the main application for such study is industrial inspection[1]. As the result, the ever-proposed algorithms have been designed to perform in ideal environments. These techniques can be adapted to less ideal environment by applying additional image processing techniques as seen in [2][3][4].

Fish detection and recognition system presented in [5] uses cascade classifier with Haar-like features. Even though providing fast and reliable detection performance, the proposed methodology of the above mentioned system requires a special setup for sample collection process. In this research, we pay our effort to develop a fish detection and recognition system that can work on non-stationary systems to be used in aquariums. Therefore, we find it difficult to implement the above methodology in our work. To overcome this difficulty, we propose the new sample collection process. Along with the improved image pre-processing pipeline and the use of the lightweight LBP (Local Binary Patterns) feature, we have made a prototype of a high accurate fish-detection system that can run at a compelling speed.

The rest of the paper is arranged as follows: In section 2, we present the related researches including cascade classifier, LBP features and fish detection system. In section 3, we present our proposal of fish detection system including the sample collection process, system framework and the image processing pipeline optimization. System evaluation is shown in section 4, followed by the conclusion.

2. Related Researches

Automatic fish detection and recognition using computer vision, even though not being an attractive research topic, has been achieved gradually for the last twenty years. Over the last decade, several works have been done to carry out the task under the less flavor environments. These methodologies are based on shape as presented in [6][7][9], or color as seen in [6][8]. The applications include fisheries management, ecological study, industrial inspection, etc.

However, the above mentioned methodologies can only be used for stationary camera to ensure the effectiveness of their segmentation models. Hence, a new approach is needed for this research. Cascade classifier [10][11] using HAAR-like features is well known for its application in the face detection. This algorithm does not require complex image preprocessing or segmentation process. With the sufficient amount of samples and training, it provides fast and reliable detection performance.

Presented in [5], the cascade classifier using Haar-like features is implemented to detect and recognize fish. In their proposal, the authors employ an automated sample collection process under the special installment. This system generates the large amount of positive samples that contain images of the fish in all possible orientations. Since those positive samples are not aligned, the classifier training requires enormous amount of time. Beside the requirement of the training time, we find it difficult to deploy the system using the same sample collection methodology.

To overcome the above mentioned difficulty, we propose the improved system that uses different sample collection process, utilizing the popular light weight LBP feature and the optimized image preprocessing pipeline. Our prototype works on samples captured by non-stationary camera with high accuracy at compelling speed.

3. System Design

3.1 Sample Collection

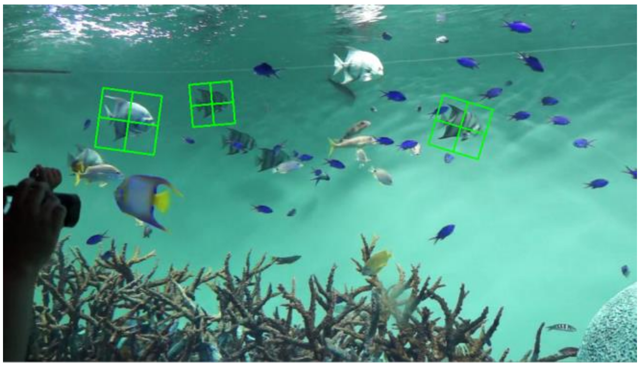

Figure 1 Positive Sample Tagging Screen
(Courtesy of Tokyo Sea Life Park)

In our system, samples are collected by manually tagging process. Using the custom software, nearly 400 samples have been able to be tagged during the time span of 1 hour. Figure 1 shows the screen of the sample tagging process. All tags are aligned with the fishes' orientation in this system. Orientation of each tag is then used to transform the corresponding sample. As the result, all positive samples are highly aligned. Because background is already included in these positive samples, no further sample generation process is required as in the standard procedure.

On the other hand, the high alignment of positive samples also mean that the system can only detect the fish at the single predefined orientation compared to the system shown in [5]. This disadvantage can be compensated by translating the orientation of the query prior to the detection process. This tradeoff is worth for considering, taking into account the ease of the sample collection process and the short training time. Furthermore, the orientation of the detected fish can be also collected by the following strategy.

3.2 Image Processing Pipeline

Image preprocessing is commonly used for Haar-like features and LBP to highlight the intensity difference between regions of the image. Under the unconstrained environment, histogram equalized grayscale image is the most ideal query. However, the stable environment inside aquarium allows us to use more sophisticated image processing model using color image.

In this model, the optimization focuses on the color of the image. Hence, the HLS color space is used since its Hue channel well reflects the main property of a color. The popular RGB color space can be also used. However, the high cost three dimensional statistic model must be used to achieve the desire performance.

The color distribution of the positive sample patch is built in this system as follows:

$$\mathrm{P(h,l,s)} \tag{1}$$

Where P is the probability of the HLS color at $\mathrm{h,l,s}$ insensitive appearing in the positive sample patch. The histogram of the hue channel is calculated by the following equation:

$$\mathrm{P(h)} = \iint \mathrm{P(h,l,s)}\, dl\, ds \tag{2}$$

To eliminate the contribution of background color to the histogram, the similar process is also done for negative sample patch. The final histogram is calculated by the following equation:

$$\mathrm{P(h)} = \mathrm{P_p(h)} - \mathrm{P_n(h)} \tag{3}$$

Where $\mathrm{P_p(h)}$ and $\mathrm{P_n(h)}$ are the hue channel distributions for positive and negative sample patches. A threshold process is applied to eliminate negative values as follows:

$$\mathrm{H(h)} = \mathrm{T(P(h),0)} = \begin{cases} \mathrm{P(h)} & \text{if } \mathrm{P(h)} > 0 \\ 0 & \text{if } \mathrm{P(h)} \le 0 \end{cases} \tag{4}$$

The histogram H is then normalized by the following equation:

$$\mathrm{H(h)} = \frac{\mathrm{H(h)} - \min\big(\mathrm{H(h)}\big)}{\max\big(\mathrm{H(h)}\big) - \min\big(\mathrm{H(h)}\big)} \tag{5}$$

After the normalization, this histogram is then used for back projection of the query image:

$$\mathrm{O(x,y)} = \mathrm{H}\big(\mathrm{I_h(x,y)}\big) \tag{6}$$

Where O is the output, I is the original query image, and $\mathrm{I_h}$ is the hue channel of the original image.

Figure 2 Processed Example of the Projection Result (Courtesy of Tokyo Sea Life Park)

Figure 2 shows a processed example of the processing pipeline. It is the merge of grayscale image and color image using the back projected image as the mask. The experiment shows that this methodology performs well on colorful fishes, but the result is worse in performance on gray-white tone (generally low saturation in HLS color space) fishes. Thus, traditional pre-processing is used when the color based processing fails to improve the performance.

To avoid the long training process due to multiple training and evaluation, a quick determination can be done by evaluating the normalized histogram. Color processing scheme is chosen if the following condition is satisfied:

$$\int \mathrm{H(h)}\, dh \ge \tau \tag{6}$$

Where τ is the predefined threshold, and $\mathrm{H(h)}$ is the normalized histogram obtained by Eq. (5).

3.3 Framework

Figure 3 shows the framework of the proposed system. In this figure, the dotted lines represent the training process and the solid lines represent the detection process.

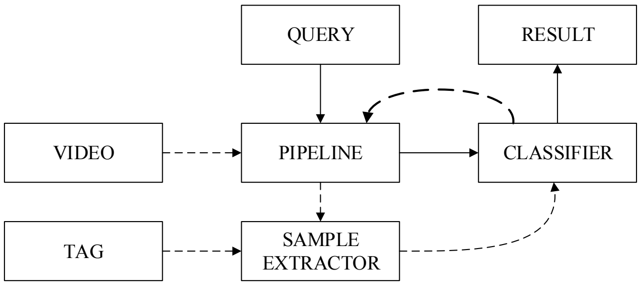

Figure 3 Framework of the Proposed System

Different from the conventional implementation, the image pre-processing pipeline in this system uses feedback from the classifier training process to determine the best model. Only two passes are required in our proposed algorithm. However, more complex color model requires longer training time along with more memory resources. Samples used for the classifier training in this system are dynamically created on demand of the pipeline training process.

4. Evaluation

4.1 System Setup

The system used for this evaluation is equipped with Intel Core i7 920 processor and 12GB of RAM. Two separated video footages taken from Tokyo Sea Life Park are used for the training and the evaluation. On each footage, 600 positive and 250 negative tags are marked for two species of fish. One species is dominated by colorful red and yellow tone while the other is dominated by gray tone.

The standard and proposed training process is then performed for each species. The same configuration is used for the cascade classifiers in the both implementations. Each cascade classifier is trained for 25 stages using the LBP feature. During the training process, we have noticed that the conventional implementation requires far more time for the training compared to the proposed implementation. This is mostly because the cluttered backgrounds are removed from the samples. Thus, it effectively simplifies the task, i.e. unburdens the training process.

4.2 Result

After the training process using the first video footage, the second video footage is used to verify the detection performance. The cascade classifier is applied on each positive as well as negative sample extracted from the testing footage. True positive is counted if the classifier reports the detection on positive sample, and false negative otherwise. Similarly, false positive is counted if the classifier reports the detection on negative sample, and true negative otherwise.

Table 1 shows the results of the evaluation. All the values are rounded averages of evaluations on two fish pieces. As shown, the proposed methodology achieves higher accuracy compared to the standard implementation. Furthermore, the new methodology requires only 48 percent of the training time for the same number of training stage compared to the standard implementation.

Table 1 Evaluation Results

	Proposed	Standard	A	B
True Positive	443	388	393	580
False Negative	157	212	207	20
False Positive	0	0	83	169
True Negative	250	250	167	81
Training Time (second)	227	472	23	17

Noted that the input is same in both implementations even though trained by two different methodologies. Because the training samples are highly aligned, the training time of the standard implementation shown in the result is already smaller compared to training using heterogeneous dataset as in [5].

Table 1 also includes the result of two additional tests. Test A is the result of fish detection using proposed processing pipeline on gray colored fish without falling back to the conventional processing method. The result shows high false positive compared to standard implementation. This is due to the low performance in the background removal at certain scenes. Figure 4 shows some examples of this case.

Test B is the standard implementation with only 15 stages of the cascade classifier training which requires the similar amount of time required for training the proposed implementation. Even though it provides the highest number of true positives, it also reports the significant number of false positive. Thus, it cannot be used for practical implementation.

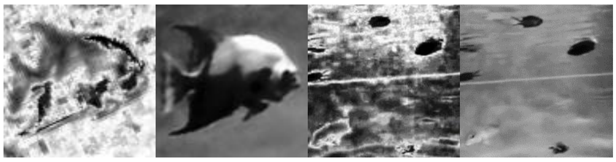

Figure 4 Examples of Bad Image Preprocessing Result (Courtesy of Tokyo Sea Life Park)

It is worth to mention that this evaluation results do not reflect the best performance of the real implementation where the number of classifier stages and the number of samples can be tweaked to achieve the best raw detection performance. Furthermore, additional layers of filters would improve the performance of detection on video feeds.

Training time in real implementation may varies depends on the scenario. A fish tank dominated by colorful species requires less time for training. Vice versa, a fish tank dominated by colorless species requires more time for training. According to our observation, fish tanks populated by only colorless species are particularly rare.

5. Conclusion

In this paper, we have presented a fish detection and recognition system using LBP cascade classifier with the optimized processing pipeline. In this system, both positive and negative samples are obtained at high rate by tagging recorded video footages using the custom software. This process does not require any special installment compared to the conventional implementations. Before feeding to the cascade classifier, query images are preprocessed using the optimized pipeline to achieve higher detection rate.

Under the stable condition of aquarium, the preprocessing pipeline optimization is proven to be effective in improving detection performance. However, as shown in Test A, there are cases that the current analysis model performs worse compared to the conventional method. Thus, more sophisticated model is needed. This issue is to be considered for our future work.

Reference

1) Yutaka N. and Takakazu I.: A Morphological Approach to Fish Discrimination. IAPR Workshop in Machine Vision Application (1998)
2) Andrew R., Greg M. and Lawrence M. Dill: One fish, two fish, butterfish, trumpeter: Recognizing fish in underwater video. IAPR Workshop in Machine Vision Application (2007)
3) Spampinato et al.: Detecting, Tracking and Counting Fish in Low Quality Unconstrained Underwater Videos. VISAPP (2008)
4) Spampinato et al.: Automatic fish classification for underwater species behavior understanding. ARTEMIS '10 pp.45-50 (2010)
5) Benson, Bridget, et al.: Field Programmable Gate Array (FPGA) Based Fish Detection Using Haar Classifiers. American Academy of Underwater Sciences (2009).
6) Strachan NJC: Recognition of fish species by color and shape. Image Vision Computing (1993)
7) Cadieux S. et al.: Intelligent system for automated fish sorting and counting. Intelligent Robots and Systems (2000)
8) Lee DJ, et al.: An automated fish species classification and migration monitoring system. IEEE Industrial Electronics Society (2003)
9) Lee DJ, et al.: Contour matching for a fish recognition and migration monitoring system. SPIE optics east (2004).
10) Paul Viola and Michael J. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE CVPR (2001)
11) Rainer Lienhart and Jochen Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. IEEE ICIP (2002)

Acknowledgments - We would like to acknowledge Tokyo Sea Life Park for letting us use the recorded materials in our research.

